VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INFORMAČNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INFORMATION SYSTEMS

DETEKCE SLOVNÍKOVÝCH ÚTOKŮ NA SÍŤOVÉ SLUŽBY ANALÝZOU IP TOKŮ

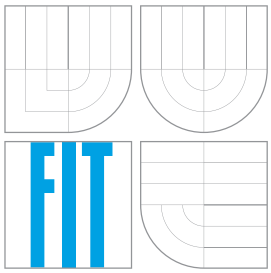
DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE
AUTHOR

Bc. MARTIN ČINČALA

BRNO 2015

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INFORMAČNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INFORMATION SYSTEMS

DETEKCE SLOVNÍKOVÝCH ÚTOKŮ NA SÍŤOVÉ SLUŽBY ANALÝZOU IP TOKŮ

DETECTION OF DICTIONARY ATTACKS ON NETWORK SERVICES USING IP FLOW ANALYSIS

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE
AUTHOR

Bc. MARTIN ČINČALA

VEDOUCÍ PRÁCE
SUPERVISOR

Ing. PETR MATOUŠEK, Ph.D.

BRNO 2015

Abstrakt

Stávající výzkumy naznačují, že je možné detekovat slovníkové útoky pomocí toků dat. Tento typ detekce byl úspěšně implementován například pro protokoly SSH, LDAP a RDP. Pro zjištění, zda je možné stejné způsoby detekce použít i pro poštovní protokoly, bylo vytvořeno virtuální testovací prostředí. Z dat, které jsem v tomto prostředí získal, se mi podařilo odvodit charakteristiky útoků v tocích a zvolit statistickou hodnotu, která útoky odliší od legitimního provozu. Za hlavní charakteristiku útoků jsem zvolil rozptyl určitých parametrů toků. IP adresy, jejichž toky mají malý rozptyl vybraných parametrů a vysokou frekvenci příchodu paketů jsou považovány za nedůvěryhodné. Aby jsme vyloučili falešné detekce, rozptyl je počítán z historie IP adresy, která v případě legitimního uživatele obsahuje různé toky a zabrání označení této IP adresy za nebezpečnou. Tento princip byl použit k vytvoření skriptu, který detekuje útoky z výstupů kolektoru nfdump. Úspěšnost detekce útoků byla testována na klasifikovaných datech z reálného prostředí. Výsledky testů ukázali, že při dobrém nastavení hraničních hodnot je procento zachycených útoků velmi vysoké a výsledky jsou bez falešných pozitivních detekcí. Detekce útoků není omezena jen na poštové protokoly. Vzhledem k tomu, že návrh je univerzální, skript dokáže detekovat slovníkové útoky na SSH, LDAP, SIP, RDP, SQL, telnet i některé další útoky.

Abstract

Existing research suggests that it is possible to detect dictionary attacks using IP flows. This type of detection was successfully implemented for SSH, LDAP and RDP protocols. To determine whether it is possible to use the same methods of detection for e-mail protocols virtual test environment was created. I deduced the characteristics of attacks in flows from the data, which I gained from this virtual environment. Than I chose the statistical value that separates the attacks from legitimate traffic. Variance of specific flow parameters was chosen as main characteristic of attacks. IP addresses with flows that have small variance of chosen parameters and high frequency of packet arrival are considered untrustworthy. Variance is calculated from IP history to rule out false positives. The IP history of legitimate user contains variation of flows which prevents marking this IP address as dangerous. On the basis of this principal the script, which detects the attacks from the nfdump output, was created. The success of detection of the attacks was tested on classificated data from the real environment. The results of tests showed, that with good configuration of marginal values the percentage of detected attacks is high and there are no false positives. Detection is not limited only on mail protocols. With regard to universal design, the script is able to detect dictionary attacks on SSH, LDAP, SIP, RDP, SQL, telnet and some other attacks.

Klíčová slova

SMTP, IMAP, POP3, netflow, slovníkové útoky

Keywords

SMTP, IMAP, POP3, netflow, dictionary attacks

Citace

Martin Činčala: Detekce slovníkových útoků na síťové
služby analýzou IP toků, diplomová práce, Brno, FIT VUT v Brně, 2015

Detekce slovníkových útoků na síťové služby analýzou IP toků

Prohlášení

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně pod vedením pana Ing. Petra Matouška Ph.D.

.....
Martin Činčala
24. mája 2015

Poděkování

Ďakujem vedúcemu práce Ing. Petrovi Matouškovi Ph.D. za rady a nápady vďaka ktorým mohla táto práca vzniknúť. Zároveň ďakujem Tomášovi Jančovi za poskytnutie prístupu k reálnym dátam, na ktorých som mohol otestovať silné stránky implementovaného skriptu. Dáta, ktoré overili slabé stránky skriptu som získal od Ing. Matěja Grégra, za čo mu tiež veľmi pekne ďakujem.

Obsah

Kapitola 1

Úvod

Monitorovanie na úrovni tokov je dnes bežný spôsob sledovania sietí. Zo získaných tokov je možné získať komplexný obraz o prevádzke celej siete. Zaťaženie siete je závislé hlavne na čase a výkyvy charakteristík od normálu často znamenajú nejaký problém v sieti, napríklad rôzne útoky. Toky obsahujú len parametre z hlavičiek paketov, nemajú informácie o obsahu paketov, preto treba rozhodnutie či ide o útok urobiť len na základe charakteristického správania útočníkov. Niektoré útoky sa napriek tomu dajú detekovať pomerne spoľahlivo. Článok [19] vysvetľuje, ktoré útoky sa dajú detekovať z tokov a ktorých detekcia je obtiažna. Článok [9] sa zaoberá implementáciou slovníkových útokov na protokol SSH a vďaka jeho kvalite bol použitý ako jeden z hlavných zdrojov pri písaní tejto práce.

V súčasnosti sú rozšírené dva spôsoby detekcie a obrany proti útokom na poštové servery: oneskorenie odpovede od serveru v prípade zadania zlého hesla a zablokovanie prístupu k účtu používateľa po definovanom počte neplatných pokusov. Tieto prístupy sú zneužiteľné útočníkom, ktorý môže zablokovať prístup k účtom ľubovoľnému počtu používateľov alebo obísť danú ochranu, pokiaľ skúša heslá na rôzne účty, prípadne na rôzne poštové servery. Detekcia útokov z tokov dát by nám umožnila detekovať a prípadne zablokovať útoky na centrálnom prvku. Takýto typ detekcie sa jednoduchšie nasadzuje, spravuje a má znalosti o všetkých poštových serveroch v danej sieti. Ďalšou výhodou je, že nezaťažuje koncové zariadenia.

Na začiatku práce sú zhrnuté základy monitorovania sietí pomocou tokov. Kapitola 3 popisuje spôsoby šifrovania komunikácie medzi klientom a poštovým serverom a vlastnosti poštových protokolov SMTP, IMAP a POP3. Kapitola 4 sa zaoberá charakteristikami tokov poštových protokolov, útokmi na tieto protokoly a existujúcimi spôsobmi detekcie slovníkových útokov. Zároveň sú v tejto kapitole predstavené prostredia, ktoré slúžili na vygenerovanie a zber dát o útokoch. Výsledky získané z týchto dát sú prezentované na konci kapitoly a použité na implementáciu skriptu, ktorý popisuje kapitola 5. Skript som otestoval vo virtuálnom a produkčnom prostredí a výsledky testov popísal v kapitole 6. V kapitole 7 sú porovnané výhody a nevýhody detekcie slovníkových útokov z tokov dát, silné a slabé stránky implementovaných skriptov a možnosti použitia vytvorených nástrojov. Zároveň som sa snažil navrhnúť, ktoré parametre paketov by pomohli detekcii útokov, ak by ich sonda exportovala. Záver kapitoly je venovaný porovnaniu vytvorených skriptov s existujúcimi nástrojmi.

Kapitola 2

Monitoring siete pomocou IP tokov

V niektorých prípadoch nie je možné monitorovať siete na úrovni paketov. Takýto typ monitoringu je buď drahý alebo úplne nemožný kvôli vysokým prenosovým rýchlostiam sietí. Sledovanie siete na úrovni tokov eliminuje tieto nevýhody a poskytuje iný pohľad na kontrolu siete. Tok je súbor paketov, ktoré prechádzajú cez monitorovací bod v sieti v určitom časovom intervale. Všetky pakety patriace k danému toku majú niekoľko spoločných vlastností [10] (minimálne zdrojovú a cieľovú IP adresu, zdrojový a cieľový port a protokol). Namiesto obsahu paketov sa analyzujú počty paketov zo zdrojovej na cieľovú IP adresu, čísla portov, čas začiatku komunikácie, dĺžka komunikácie, množstvo prenesených dát a ďalšie informácie, ktoré môžeme získať z tokov. Príklad toku zobrazuje výpis 2.1.

2.1 Architektúra

Na zber štatistík sa používajú dve zariadenia - sonda a kolektor. Sonda monitoruje dátový tok na vybranom rozhraní sieťového prvku (často hraničnom smerovači) a štatistiky si ukladá do cache pamäte. Expiráciu záznamov v cache a ich následné odoslanie na kolektor určujú dva timeout časovače - pasívny a aktívny.

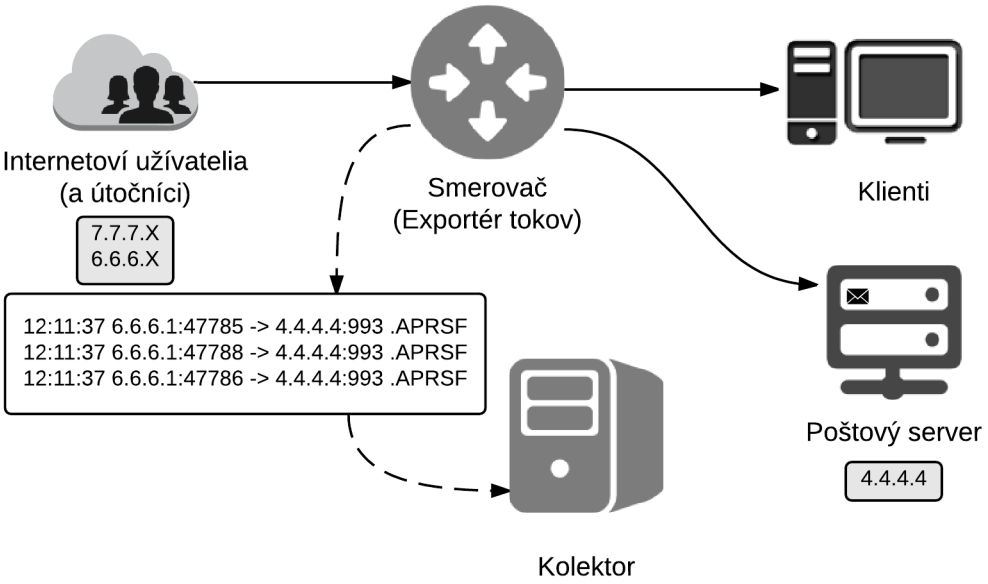
Pasívny timeout nastane ak sonda neprijala žiadne pakety, ktoré patria k danému toku za stanovený časový interval (obvykle 15 sekúnd až 5 minút)

Aktívny timeout - ak tok trvá dlhšie ako je definovaný aktívny timeout, expirujú informácie o tomto toku. Zaisťuje, že kolektor dostáva periodické informácie o veľmi dlhých tokoch.

Po vypršaní niektorého z timeout časovačov odošle sonda informácie o tokoch kolektoru, ktorý ich predspracováva a ukladá, aby mohli byť analyzované.

Na obrázku 2.1 je jeden z možných spôsobov zapojenia sondy a kolektora. Sonda (flow exporter) je súčasťou hraničného smerovača. Obrázok obsahuje IP adresy, ktoré sú použité v ďalších kapitolách. Anonymizovaná IP adresa poštového serveru je 4.4.4.4, pre útočníkov budem používať 6.6.6.X a pre legitímnych používateľov 7.7.7.X.

Na zníženie záťaže siete a kolektora môžu byť pakety pred odoslaním na kolektor vzorkované. Vzorky môžu byť vyberané systematicky alebo náhodne. Pri systematickom vzorkovaní je výber paketov deterministický (napríklad každý desiaty paket). Náhodné vzorkovanie využíva metódu, ktorej výstupom je náhodná množina paketov. Hlavným cieľom je vybrať reprezentatívnu podmnožinu paketov. Použitie systematického vzorkovania môže



Obr. 2.1: Topológia zapojenia sondy a kolektora

mať za následok vznik neželaných korelácií a skreslení pri periodickej sieťovej prevádzke, preto sa väčšinou preferuje náhodné vzorkovanie [7].

Na výber paketov s určitou vlastnosťou sa používa filtrovanie paketov. Rovnako ako vzorkovanie znižuje záťaž siete a kolektora a znova ho môžeme rozdeliť na dve skupiny.

Filtrovanie podľa vlastnosti (Property Match Filtering) Paket je vybraný, ak sa jeho určité vlastnosti zhodujú so zvolenými hodnotami, napríklad filtrovanie podľa IP adresy, portov atď.

Filtrovanie podľa hašu (Hash-Based Filtering) Aplikuje hašovaciu funkciu na paket alebo jeho časť a podľa výsledku zahodí alebo vyberie paket. Tento typ filtrovania je efektívny ak vyberáme pakety podľa viacerých parametrov [7].

Výpis 2.1 zobrazuje toky z kolektoru nfdump[1], ktorého výstupy sú použité pre detekciu útokov v tejto práci. Prvý parameter označuje čas príchodu prvého paketu v danom toku. Za ním nasleduje dĺžka trvania toku, zdrojová/cieľová IP adresa a port, počet paketov v toku, celkový počet prenesených bajtov a počet tokov, ktoré boli do daného záznamu agregované. Z týchto údajov je možné vypočítať ďalšie štatistiky - napríklad priemerný počet bajtov na paket, počet paketov za sekundu alebo počet bajtov za sekundu.

Listing 2.1: Príklad tokov (výstup z nfdump)

```
Date first seen          Duration Proto Src IP:Port      Dst IP:Port   Packets Bytes Flows
2014-11-17 16:18:39.728     0.171 TCP   7.7.7.1:45154 -> 4.4.4.4:993        18  2344     1
2014-11-17 16:18:39.729     0.171 TCP   4.4.4.4:993   -> 7.7.7.1:45154      15  2430     1
```

[1] http://nfdump.sourceforge.net/

2.2 NetFlow a IPFIX

Formát dát, v ktorom sa toky exportujú na kolektor, špecifikujú v súčasnosti dva protokoly - NetFlow [2] a IPFIX [10]. NetFlow je otvorený protokol navrhnutý spoločnosťou Cisco Systems na monitorovanie sieťovej prevádzky pôvodne pre Cisco smerovače. Napriek tomu, že k nemu nebola vydaná oficiálna dokumentácia, jeho rozsiahle používanie prispelo k tomu, že Cisco poskytlo tento formát voľne k dispozícii [1]. Okolo roku 2002 bola vydaná verzia NetFlow v5. Tú neskôr nahradil NetFlow v9, ktorý pridal podporu šablón, IPv6, VLAN a ďalších funkcií.

V roku 2004 začalo IETF vytvárať štandard pre export dátových tokov IPFIX [3]. Z kandidátnych protokolov (CRANE, Diameter, LFAP, NetFlow v9, Streaming IPDR) bol ako základ pre nový štandard zvolený NetFlow v9 kvôli jeho rozšírenosti a jednoduchosti [13]. Protokol IPFIX definuje formát tokov, pravidlá pre implementáciu a riešenie špeciálnych prípadov, napríklad obojsmerné toky, odstránenie redundancie, agregáciu alebo anonymizáciu tokov.

2.3 Dostupné sondy a kolektory

Sondy a kolektory môžeme rozdeliť do dvoch skupín: hardvérové a softvérové. Hardvérové sondy dosahujú lepšiu priepustnosť, softvérové sondy sú flexibilnejšie - môžu byť nasadené na rôznom hardvéri a poskytovať viac špecifických služieb. Softvérové sondy sú zároveň lacnejšie. Ďalšími dôležitými parametrami sú veľkosť cache, aby sonda zvládla 'držať v pamäti' veľký počet tokov súčasne a podpora IE (Informačný element). Staršie sondy dokážu často exportovať len obmedzenú množinu IE, novšie sondy pridávajú export MAC adries, VLAN tagov, MPLS návestí alebo aplikačných dát.

Dostupné open-source sondy sú napríklad: fprobe, nProbe, pmacct, softflowd a Vermont. Všetky tieto sondy podporujú NetFlow v5, NetFlow v9 a IPFIX, aj keď jeho podpora je často iba čiastočná - väčšinou chýba podpora štrukturovaných dát a šablón. Viac informácií o dostupných sondách a prehľadné porovnanie ich vlastností obsahuje článok [7].

Komerčné sondy sa predávajú buď ako dedikované hardvérové sondy alebo sú súčasťou smerovačov, prepínačov a firewallov. Niektorí výrobcovia majú vlastnú implementáciu NetFlow, ale po štandardizácii IETF sa väčšina prikláňa k IPFIX štandardu. Medzi najznámejšie komerčné sondy patrí Netflow Generation Appliance (Cisco), FlowMon Probe (INVEA-TECH) a nBox (ntop). Prehľadná tabuľka komerčných sond, prepínačov, smerovačov a firewall-ov, ktoré umožňujú exportovať toky je opäť v článku [7].

Kolektory prijímajú a spracovávajú toky dát od jednej alebo viacerých sond. Základnými úkonmi nad dátami je ich kompresia, agregácia, anonymizácia, filtrovanie a vygenerovanie výstupov. Spracované dáta môžu byť uložené v súboroch alebo v databázi. Pri ukladaní dát do súborov nepotrebujeme indexy, ktoré zaberajú pamäťový priestor a je ich nutné generovať. Naviac sú súbory ľahšie čitateľné pre väčšinu nástrojov. Výhody uloženia dát do databázy sú rýchlejšie vyhľadávanie a možnosť použitia vyššej kompresie, pretože údaje v stĺpcoch sú väčšinou veľmi podobné. Kolektor nfdump, ktorý budeme používať v tejto práci ukladá informácie do súborov. Okrem ukladania dát sú dôležitými parametrami kolektorov aj výkonnosť (počet tokov, ktorý kolektor dokáže spracovať za sekundu) a podporované IE.

Príklady open-source kolektorov: flowd, nfdump, nProbe, pmacct, SiLK. Všetky vymenované kolektory podporujú NetFlow v5 a NetFlow v9. Podpora IPFIX parametrov je rôzna. Obojsmerné toky podporoval v čase písania tohto textu len nProbe a pmacct, šablóny podporujú všetky kolektory okrem flowd a štrukturované dáta podporuje len SiLK[7].

Komerčné kolektory vyrába napríklad Arbor Networks Fluke Networks, INVEA-TECH, Lancope alebo Solar-Winds. Všetky spomenuté kolektory podporujú NetFlow v5, NetFlow v9 a IPFIX štandard. Predávajú sa väčšinou ako hotové hardvérové zariadenia a ich výkonnosť sa udáva ako počet spracovaných tokov za sekundu. Pre porovnanie Arbor Networks dokáže spracovať 250 000 tokov/s, výkonnosť kolektora od INVEA-TECH sa pohybuje od 75 000 do 200 000 tokov/s a Fluke Networks spracuje 40 000 tokov/s. Pri použití virtuálneho kolektora sľubuje INVEA-TECH výkonnosť 200 000 tokov/s pre server s minimálnou konfiguráciou 8 CPU jadier a 16 GB pamäte RAM [8].

Kapitola 3

Protokoly elektronickej pošty

Komunikáciu medzi e-mailovými klientami a servermi riadia poštové protokoly. Najpoužívanejšie protokoly sú IMAP, POP3 a SMTP. IMAP a POP3 slúžia na príjem správ, SMTP na prenos a odosielanie. Táto kapitola popisuje spôsoby, ako býva najčastejšie šifrovaná komunikácia týchto protokolov a vlastnosti troch najpoužívanejších protokolov. Z vlastností protokolov je pre túto prácu dôležitá hlavne fáza autentizácie.

3.1 Šifrovanie komunikácie

Komunikácia so serverom môže byť zabezpečená pomocou protokolov SSL alebo TLS. Detailnejší popis týchto protokolov nie je predmetom tejto práce. Zjednodušene sa dá povedať, že SSL je predchodca TLS protokolu a boli v ňom opravené niektoré bezpečnostné zraniteľnosti, ktoré obsahoval protokol SSL. Každý z poštových protokolov má okrem štandardného vyhradený ďalší port, cez ktorý komunikuje s klientom šifrovane. Prehľad portov je v tabuľke 3.1.

Protokol	Nešifrovaná komunikácia / STARTTLS	SSL / TLS
POP3	110	995
SMTP	25 a 587	465
IMAP	143	993

Tabuľka 3.1: Porty poštových služieb

Aj na porte určenom pôvodne pre nešifrovanú komunikáciu je možné komunikovať šifrovane pomocou protokolu STARTTLS. Po nadviazaní spojenia STARTTLS vyjedná parametre a povýši spojenie na šifrované. Výhodou tohto prístupu je, že neplytvá portami na e-mailovom serveri a zjednodušuje konfiguráciu klienta, pretože sa používa ten istý port pre šifrovanú aj nešifrovanú komunikáciu. S nasadením STARTTLS sa vyskytlo niekoľko problémov. Porty pre šifrovanú komunikáciu často nie je možné zakázať z dôvodu kompatibility klientov a niektorí klienti zlyhávali počas vyjednávania zabezpečeného pripojenia. Administrátori zase zakazovali nešifrované porty, čím sa úplne vylúčilo použitie STARTTLS. Výnimkou je protokol SMTP, ktorý sa začal okrem prenosu správ používať na odovzdávanie správ serveru. Odovzdávané správy môžu byť nekompletné (chýba dátum, plne kvalifikovaný názov domény alebo iná položka hlavičky) [6]. Pre odovzdávanie správ na server bol pridelený port 587. Použitie tohto portu výslovne nevyžaduje šifrovanie, ale kvôli bezpečnosti sa často používa práve STARTTLS [5].

3.2 IMAP

IMAP je najmladší z trojice spomínaných protokolov. Klientovi poskytuje prostriedky na manipuláciu so správami, ktoré sú, narozdiel od protokolu POP3, uložené na serveri. IMAP podporuje aj offline mód, v ktorom stiahne správy zo serveru, aby k nim bolo možné pristupovať bez nutnosti pripájať sa k serveru. Pracuje nad protokolom TCP a počúva na portoch 143 a 993 (viz. tabuľka 3.1). Komunikácia prebieha spôsobom klient-server. Klient posiela príkazy, na ktoré server odpovedá príslušnými dátami a výsledkom príkazu. Po nadviazaní spojenia so serverom môže byť IMAP spojenie v jednom z piatich stavov [4]:

1. **Initial** - V tomto stave pošle server uvítaciu správu klientovi.
2. **Not Authenticated** - Klient musí poslať prihlasovacie údaje predtým, ako začne vykonávať ďalšie akcie. Ak bol klient overený vopred, tento stav sa vynechá.
3. **Authenticated** - Klient je overený, musí vybrať poštovú schránku ku ktorej chce pristupovať.
4. **Selected** - V tomto stave môže klient vykonávať akcie nad správami v poštovej schránke.
5. **Logout** - Spojenie bude ukončené. Najčastejšie nastáva, ak klient pošle LOGOUT správu, na ktorú server odpovedá správou BYE.

Keďže sa táto práca zaoberá detekciou útokov, je pre nás dôležitý stav **Not Authenticated**, v ktorom sa nachádza útočník pred uhádnutím hesla. V tomto stave môže klient zaslať tri typy správ:

1. **STARTTLS** začne vyjednávanie parametrov TLS spojenia. Klient nesmie posielať ďalšie príkazy až do skončenia vyjednávania TLS.
2. **AUTHENTICATE** príkaz určuje, ktorý SASL [1] autentizačný mechanizmus sa použije. Ak ho server podporuje, je tento autentizačný mechanizmus použitý na autentizáciu a identifikáciu klienta. Voliteľne môže byť vyjednaná vrstva na zabezpečenie neskoršej komunikácie. Vyjednávanie prebieha formou špecifických výziev od serveru a odpovedí od klienta. Servery a klienti môžu podporovať viacero autentizačných mechanizmov. Klient pošle serveru správu CAPABILITY, server vráti zoznam podporovaných autentizačných mechanizmov, z ktorých si klient môže vybrať vhodný mechanizmus na autentizáciu.
3. **LOGIN** Nešifrovane identifikuje klienta jeho menom a heslom voči serveru. Ak nebolo vyjednané žiadne zabezpečenie spojenia, server by mal zabrániť klientovi odoslať LOGIN príkaz, aby za zabránilo odchyteniu údajov treťou stranou.

[1] SASL popisuje metódy implementácie autentizácie. Protokol musí obsahovať príkaz pre identifikáciu a autentizáciu užívateľa voči serveru a voliteľne pre zabezpečenie neskoršej komunikácie. Ak je vyjednané zabezpečenie komunikácie, medzi spojenie a protokol je vložená bezpečnostná vrstva [14].

Ukážka komunikácie klienta s IMAP serverom:

```
C: openssl s_client -connect mail.example.com:993
S: * OK [CAPABILITY IMAP4rev1 LITERAL+ SASL-IR LOGIN-REFERRALS ID ENABLE
     IDLE AUTH=PLAIN AUTH=LOGIN]
C: a login testuser testuser
S: a OK [CAPABILITY IMAP4rev1 LITERAL+ SASL-IR] Logged in
C: e logout
S: * BYE Logging out
   e OK Logout completed.
```

3.3 SMTP

Protokol SMTP zabezpečuje prenos správ. Užívateľský softvér, ktorý spracováva správy sa označuje MUA (Mail User Agent). Doručovanie správ do poštovej schránky adresáta zabezpečuje MTA (Mail Transfer Agent). SMTP pôvodne definoval pravidlá len na doručenie správy do poštovej schránky adresáta. Neskôr sa začal používať aj na odovzdávanie správ serveru, ktorý doplní potrebné informácie v hlavičke. Tento typ správania zabezpečuje MSA (Mail Submission Agent). SMTP väčšinou pracuje nad TCP, ale môže byť použitý akýkoľvek transportný protokol, ktorý zachová poradie dát.

Na ustanovenie spojenia pošle klient správu serveru, ktorý mu odpovie uvítacím pozdravom. Po prijatí pošle klient EHLO správu ktorou oznamuje, že je schopný spracovať požiadavky serveru a identifikuje sa. Starší klienti, ktorí nepodporujú rozšírenia serveru, môžu poslať správu HELO.

Jedným z rozšírení SMTP protokolu je autentizácia. Klient sa autentizuje voči serveru jedným z mechanizmov, ktoré mu server poskytuje. Autentizácia je povinná len pre servery, ktoré komunikujú s MSA. Autentizáciu začína klient zaslaním správy AUTH. Parameter správy je jeden z autentizačných mechanizmov (napr. PLAIN, LOGIN alebo CRAM-MD5) [12].

AUTH PLAIN Ak server prijme AUTH PLAIN od klienta, pošle odpoveď 334. Klient potom pošle užívateľské meno a heslo vo formáte BASE64. Meno a heslo je zakódované v jednom reťazci [11].

AUTH LOGIN Komunikácia prebieha podobne ako pri AUTH PLAIN, ale meno a heslo sa posiela serveru oddelene.

AUTH CRAM-MD5 Narozdiel od predchádzajúcich mechanizmov posiela informácie zašifrované pomocou MD5. Vyjednanie parametrov prebieha spôsobom výzva-odpoveď.

Ukážka komunikácie klienta so SMTP serverom:

```
C: openssl s_client -connect mail.example.com:587 -starttls smtp
S: 250 DSN
C: EHLO mail.example.com
S: 250-mail.example.com
   250-AUTH PLAIN LOGIN
   250 DSN
C: AUTH LOGIN
```

```
S: 334 VXNlcm5hbWU6
C: dGVzdHVzZXJAZXhhbXBsZS5jb20= (testuser@example.com v BASE64 kódovaní)
S: 334 UGFzc3dvcmQ6
C: dGVzdHVzZXI=      (heslo v BASE64 kódovaní)
S: 235 2.7.0 Authentication successful
C: QUIT
S: 221 2.0.0 Bye
```

3.4 POP3

POP3 sa používa na sťahovanie správ z poštového serveru do schránky používateľa. Pracuje nad TCP a využíva porty 110 a 995 (viz. tabuľka 3.1). POP3 neposkytuje skoro žiadne možnosti manipulácie so správami na serveri. Po stiahnutí sú správy väčšinou zo serveru odstránené. Po ustanovení spojenia s POP3 serverom a prijatí uvítacej správy sa klient musí autentizovať. Klient môže použiť viacero spôsobov autentizácie.

Jedným zo spôsobov sú príkazy USER a PASS. Klient najskôr pošle príkaz USER a ako argument použije názov poštovej schránky. Server overí, či daná schránka existuje a pošle odpoveď. V prípade úspechu klient pokračuje príkazom PASS, ktorého argument je užívateľove heslo.

Ďalšou možnosťou autentizácie je použiť príkaz APOP ktorý poskytuje aj ochranu proti replay útokom a zasiela heslo šifrovane. APOP má 2 parametre - užívateľské meno, ktoré ma rovnakú funkcionalitu ako príkaz USER a parameter digest. Digest sa vypočíta aplikovaním MD5 algoritmu na reťazec obsahujúci časovú značku od serveru a zdieľané heslo. Zdieľané heslo musí poznať len konkrétny server a klient [15]. POP3 môže rovnako ako IMAP využiť autentizačný mechanizmus [18].

Ukážka komunikácie klienta s POP3 serverom:

```
C: openssl s_client -connect mail.example.com:110 -starttls pop3
S: +OK Dovecot ready.
C: user testuser
S: +OK
C: pass testuser
S: +OK Logged in.
C: quit
S: +OK Logging out.
```

Kapitola 4

Detekcia slovníkových útokov pomocou IP tokov

4.1 Slovníkové útoky

Slovníkové útoky sa dajú rozdeliť na dve skupiny: offline a online. Pri online útoku musí útočník čakať po každom pokuse o uhádnutie hesla na odpoveď od serveru. Offline útok nevyžaduje odpoveď od serveru, pretože pristupuje priamo k súboru s heslami a teda nie je detekovateľný na sieťových zariadeniach. Táto práca sa zaoberá len online útokmi.

Na detekciu slovníkových útokov na koncových zariadeniach je možné použiť nástroje, ktoré prehľadávajú log súbory, prípadne blokujú IP adresy útočníkov - napríklad Fail2ban[1], Logwatch[2] alebo IPS nástroj Snort[3], ktorý pomocou pravidiel pre konkrétne protokoly popisuje sieťovú prevádzku vo forme tokov.

Proti slovníkovým útokom existujú na koncových zariadeniach dve hlavné metódy obrany.

Oneskorenie odpovede Server oneskorí odpoveď áno/nie po prijatí užívateľského mena a hesla. Toto útočníka spomalí v skúšaní hesiel.

Zamknutie účtu Po niekoľkých neúspešných pokusoch sa účet uzamkne a nebude možné sa k nemu prihlásiť.

Tieto opatrenia obmedzujú užívateľov, zvyšujú ceny služieb a pri veľkom počte serverov nemusia byť účinné. Zamknutie účtu aj oneskorenie odpovede zlyhá v prípade ak útočníka nezaujíma konkrétny účet, ale skúša heslá voči viacerým účtom. V takom prípade môže pristupovať k veľa účtom paralelne bez toho, aby niektorý vyskúšal viackrát a bol zablokovaný. Tento prístup býva veľmi častý, hlavne za účelom rozosielania spamu. Ak útočník vyskúša štatisticky najpoužívanejšie heslá voči veľkému počtu účtov, zvýši sa tým pravdepodobnosť uhádnutia hesla.

Zablokovanie účtu často spôsobuje problémy koncovým užívateľom. Útočník môže cielene zablokovať ľubovolný počet emailových účtov ak bude opakovane zadávať neplatné heslá a blokovať tak prístup legitímnych užívateľov k ich poštovým schránkam. Organizáciám, ktoré blokujú účty sa zvýšia náklady na podporu, pretože musia odpovedať klientom,

[1] http://www.fail2ban.org/
[2] http://www.logwatch.org/
[3] https://www.snort.org/

ktorí sa nemôžu prihlásiť na svoje účty. Viaceré spoločnosti (napr. eBay a Yahoo) zrušili kvôli týmto dôvodom blokovanie prihlasovania k poštovým účtom[16].

Detekcia útokov na sieťovej vrstve má oproti detekcii na koncových zariadeniach niekoľko výhod: nezaťažuje koncové zariadenia, detekcia je centrálna – často stačí jedno zariadenie pre celú sieť, dokáže lepšie detekovať distribuované útoky – ak útočník rozloží útok medzi viacero serverov v sieti, útok nemusí byť na koncovom zariadení detekovaný.

4.2 Existujúce riešenia

Detekcia slovníkových útokov na SSH server - 1. prístup

Článok [9] popisuje detekciu slovníkových útokov na SSH server z tokov dát. Autori nasadili netflow sondy a zbierali štatistické dáta po dobu 30 dní. Za tento čas detekovali 911 pokusov o útok a vyvodili z nich nasledujúce charakteristiky slovníkových útokov v tokoch:

- TCP port obete je 22, TCP port útočníka náhodný a väčší ako 1024
- veľa tokov (stovky až tisíce) od útočníka smerom k obeti v krátkom časovom intervale (5 minút)
- toky od útočníka majú malú veľkosť: 10 až 30 paketov a 1400 až 5000 bajtov
- odpovede od obete sú tiež malé (často majú rovnaký počet paketov a bajtov)
- dĺžka toku je až do 5 sekúnd
- posledný tok je v prípade úspešného útoku odlišný

Podľa testov (prihlásenie na SSH, kopírovanie súborov cez SCP, sťahovanie veľkých súborov cez SFTP a rsync) sa tieto charakteristiky nevyskytujú v bežnej sieťovej prevádzke SSH protokolu. Zároveň tieto štatistiky vyhovujú všetkým simulovaným útokom, ktoré autori uskutočnili.

Algoritmus detekcie bol implementovaný pomocou rozhodovacieho stromu, ktorý sa ľahko prispôsobuje meniacim sa charakteristikám sietí. Atribúty rozhodovacieho stromu sú zdrojová a cieľová IP adresa, čas začiatku toku, dĺžka toku, počet paketov a bajtov. Aby sa zabránilo falošným detekciám, počet paketov je obmedzený zdola a počet bajtov zhora zvolenou konštantou. Záznamy tokov sú vyhodnocované rozhodovacím stromom, ktorý má na začiatku pevne nastavené hranice atribútov. Pre každý pár (útočník, obeť) sa vytvorí pole parametrov, ktoré je upravované tolerančnými faktormi až pokiaľ sa nedosiahne hranica potrebná k označeniu záznamu za útok. Pre daný pár (útočník, obeť) sú upravené parametre rozhodovacieho stromu a nasledujúce útoky sa detekujú podľa týchto nových parametrov. Rozhodovací strom obsahuje dva typy parametrov - statické a dynamické. Statické parametre určujú citlivosť útoku a nastavujú sa ručne. Dynamické parametre sa menia za behu podľa charakteristík tokov.

- Statické parametre:
  - počet pokusov z 1 IP adresy na vyhlásenie útoku
  - počet pokusov z viacerých IP adries na vyhlásenie útoku
  - maximálny čas medzi dvoma pokusmi o útok z jedného zdroja
  - počiatočné hodnoty počtu paketov, bajtov a trvania toku

  - počiatočné hodnoty tolerančných faktorov (počet paketov, bajtov, trvanie toku)
  - dĺžka histórie tokov, z ktorej sa detekuje útok
- Dynamické parametre:
  - počet útokov od konkrétneho útočníka/na konkrétnu obeť
  - čas posledného pokusu o útok
  - počet paketov, počet bajtov, dĺžka toku pre každý pár (útočník, obeť)

Na bežne dostupnom hardvéri dokáže algoritmus spracovať približne 2500 tokov za sekundu. Táto hodnota je dostatočná pre použitie na 10 Gb/s sieťových rozhraniach v sieťach s tisíckami počítačov. Algoritmus detekoval všetky útoky, ktoré boli odhalené z log súborov SSH serveru. Pri testovaní počas 23 dní bolo zaznamenaných 65 útokov, 1 útok nebol detekovaný [21]. Detailnejšie štatistiky o úspešnosti detekcie a počte falošných detekcií článok neuvádza.

Detekcia slovníkových útokov na SSH server - 2. prístup

V ďalšom článku [17] autori skúmali detekciu slovníkových útokov na SSH server pomocou dvoch kritérií:

Existencia komunikačného kanálu – určuje, či bol medzi útočníkom a klientom vytvorený komunikačný kanál a tým detekuje úspešnosť útoku

Interval medzi príchodom autentizačného paketu – určuje, či meno a heslo napísal užívateľ

Kvôli zabezpečeniu SSH protokolu tieto štatistiky nie sú priamo prístupné. Autori ich získavajú analýzou tokov. Porovnávajú počty, veľkosti a časy príchodu paketov v rôznych fázach pripojenia (počas autentizácie a po úspešnej autentizácii v priebehu komunikácie). Úspešnosť detekcie útokov na obmedzenom počte testovacích dát bola 95%. Tento typ detekcie útokov je ale ťažké implementovať, pretože súčasné sondy neposielajú potrebné údaje (napríklad časy príchodov jednotlivých paketov). Meno a heslo pre prihlásenie na poštové protokoly často nezadáva užívateľ, ale je uložené v databázi MTA, čo znemožňuje aplikáciu druhého kritéria a použitie tohto prístupu pre detekciu útokov na poštové protokoly.

Detekcia slovníkových útokov na autentizáciu RDP

Martin Vizváry popisuje vo svojej diplomovej práci [20] detekciu slovníkových útokov na autentizáciu RDP. Útoky sú detekované z tokov dát, ktoré sú získavané z výstupov kolektora nfdump. Na odlíšenie útokov od legitímnej sieťovej prevádzky používa autor filter s nasledujúcimi metrikami:

- počet prichádzajúcich paketov: <20, 100 >
- objem prichádzajúcej sieťovej prevádzky v bajtoch: <2200, 8001>
- počet odchádzajúcich paketov: <30, 190>
- objem odchádzajúcej sieťovej prevádzky v bajtoch: <3000, 180000>
- TCP príznaky ACK, PUSH, RESET, SYN

- adresa lokálnej monitorovanej siete

Ak konkrétny tok vyhovuje týmto pravidlám, je označený za útok. Autor testoval nástroj v sieti Masarykovej univerzity. Z viac ako 13 000 unikátnych IP adries vyhodnotil 3430 ako útok. Počty skutočných útokov a mieru falošne detekovaných útokov autor neuvádza. Zhodnocuje ale, že počet detekovaných útokov je nižší, pretože nie všetci útočníci splnili podmienky útoku a časť útokov nevyhovela metrikám (počet prenesených paketov a objem dát).

Detekcia slovníkových útokov na autentizačné služby (LDAP)

Detekciu slovníkových útokov na autentizačné služby skúmal Radek Šembera. Vo svojej práci [22] sa venuje hlavne detekcii útokov na LDAP . Navrhnutá metóda počíta percentuálny podiel počtu tokov a bajtov pre konkrétnu IP adresu ku všetkým tokom smerujúcim na daný LDAP server. Ak má tok objem dát menší ako 650 B alebo trvá dlhšie ako 20 sekúnd, je označený za podozrivý. Pravdepodobnosť útoku pre danú stanicu sa počíta ako súčin percentuálneho zastúpenia stanice na celkovej komunikácii a percentuálnej hodnoty pravdepodobnosti podozrenia na útok. Skript bol otestovaný v sieti Masarykovej Univerzity, útoky sa generovali pomocou penetračných testovacích programov. Skript zachytil 37 útokov, z čoho 30 bolo vygenerovaných útokov a 7 falošných detekcií. Falošné detekcie spôsobili servery, ktoré vykonávajú pravidelné dotazy na LDAP server. Keďže tieto dotazy boli veľmi podobné vygenerovaným útokom, autor odporúča pridať dané servery do 'whitelist'-u.

4.3 Testovacie prostredia

Na zozbieranie dát potrebných k návrhu algoritmu detekcie boli vytvorené dve testovacie prostredia. V každom z nich je nasadená sonda s kolektorom na zber tokov dát a poštový server.

4.3.1 Virtuálne prostredie

Pre získanie presnejších charakteristík útokov bolo nutné vytvoriť prostredie, na ktoré je možné neobmedzene útočiť. Na tento účel bol použitý virtualizačný nástroj VirtualBox [4], v ktorom boli inštalované dva virtuálne stroje. Prvým je FlowMon Collector od spoločnosti INVEA-TECH[5], ktorý obsahuje sondu a kolektor. Virtuálna sieťová karta FlowMon sondy je v promiskuitnom režime, aby mohla zachytávať všetky pakety. Druhým virtuálnym strojom je poštový server, na ktorom bežia služby SMTP, POP3 a IMAP. SMTP zabezpečuje Postfix [6], POP3 a IMAP Dovecot [7]. Poštový server je plne funkčný, aby bolo možné simulovať reálnu prevádzku. Útoky na tento server sú generované z podkladového počítača, nad ktorým bežia virtuálne stroje.

Útoky boli generované pomocou programov Hydra [8] a Medusa [9]. Príklady útokov a charakteristiky tokov popisuje kapitola 4.4.

[4] https://www.virtualbox.org/
[5] https://www.invea.com/
[6] http://www.postfix.org/
[7] http://www.dovecot.org/
[8] https://www.thc.org/thc-hydra/
[9] http://foofus.net/goons/jmk/medusa/medusa.html

4.3.2 Reálne prostredie

Na získanie dát z reálnej provádzky bola inštalovaná sonda a kolektor do serverovne s existujúcimi poštovými servermi. Ako kolektor bol použitý nfdump inštalovaný na virtuálnom FreeBSD serveri. Sondu zabezpečoval najskôr hraničný smerovač Mikrotik, ten mal ale chybu v softvéri a nezobrazoval správne dĺžky tokov, ktoré trvali kratšie ako 1 sekundu. Sonda bola preto nahradená sondou fprobe na poštovom serveri. Aktívny a neaktívny timeout bol nastavený na 60, resp. 15 sekúnd. Na poštovom serveri bol nainštalovaný operačný systém Debian. Softvér Postfix bol použitý ako SMTP server, IMAP a POP3 zabezpečoval Dovecot. Na poštovom serveri boli desiatky aktívnych poštových účtov. Aby sme mohli klasifikovať toky, bol na kolektore inštalovaný Syslog server, ktorý dostával správy o neúspešných prihláseniach na IMAP a POP3 na poštovom serveri.

4.4 Charakteristiky tokov poštových protokolov

Táto podkapitola obsahuje príklady tokov štandardnej sieťovej prevádzky a útokov. Príklady sú zobrazené vo dvojiciach - prvý výpis obsahuje vždy toky smerom k poštovému serveru, druhý výpis obsahuje opačný smer (zjednodušene odpovede serveru). Dôvodom rozdelenia výpisov je zvýšenie prehľadnosti, pretože v spoločnom výpise nebolo jednoznačne vidieť opakujúce sa charakteristiky tokov. Výpisy 4.1 a 4.2 zobrazujú štandardnú prevádzku protokolu IMAP. Výpisy 4.3 a 4.4 zobrazujú útok na IMAP server. Každý tok zodpovedá jednému pokusu o prihlásenie. Parametre dátum, TOS a počet tokov boli z priestorových dôvodov z výpisov odstránené. Tieto štatistiky sú pre všetky zobrazené toky rovnaké, pričom počet tokov je vždy 1. Nasledujúce výpisy slúžia na vysvetlenie, ktoré parametre tokov sa dajú použiť na detekciu útokov.

```
0  First seen    Duration Src Addr:Port          Dst IP:Port          Flags Pac  Bytes pps    bps  Bpp
1  00:17:29.612    1.439 7.7.7.1:53810 -> 4.4.4.4:993   .AP.S. 28   2935  19  16316  104
2  00:17:30.572    1.481 7.7.7.1:53814 -> 4.4.4.4:993   .AP.S. 35   3623  23  19570  103
3  00:23:19.321    0.900 7.7.7.1:53810 -> 4.4.4.4:993   .APR.F  8    614   8   5457   76
4  00:23:19.325    0.897 7.7.7.1:53814 -> 4.4.4.4:993   .APR.F 10    780  11   6956   78
5  01:31:41.265    1.713 7.7.7.1:53808 -> 4.4.4.4:993   .AP.S. 36   3675  21  17162  102
6  01:31:39.655    2.145 7.7.7.1:53801 -> 4.4.4.4:993   .AP.S. 28   2935  13  10946  104
7  01:39:36.755    0.072 7.7.7.1:53801 -> 4.4.4.4:993   .APR.F  3    218  41  24222   72
8  01:39:36.696    0.108 7.7.7.1:53808 -> 4.4.4.4:993   .APR.F  5    384  46  28444   76
9  07:18:29.686   27.235 7.7.7.1:53874 -> 4.4.4.4:993   .APRSF 33   3331   1    978  100
10 07:18:31.138   25.779 7.7.7.1:53879 -> 4.4.4.4:993   .APRSF 46   4381   1   1359   95
```

Listing 4.1: Štandardná sieťová prevádzka IMAP serveru (prichádzajúce toky)

```
0  First seen    Duration Src Addr:Port          Dst IP:Port          Flags Pac  Bytes pps    bps  Bpp
1  00:17:29.612    1.388 4.4.4.4:993 -> 7.7.7.1:53810   .AP.S. 18   4704  12  27112  261
2  00:17:30.572    1.428 4.4.4.4:993 -> 7.7.7.1:53814   .AP.S. 28  18825  19 105462  672
3  00:23:19.321    0.900 4.4.4.4:993 -> 7.7.7.1:53810   .APR.F  7    426   7   3786   60
4  00:23:19.325    0.897 4.4.4.4:993 -> 7.7.7.1:53814   .APR.F  9    555  10   4949   61
5  01:31:41.265    1.659 4.4.4.4:993 -> 7.7.7.1:53808   .AP.S. 29  18914  17  91206  652
6  01:31:39.655    2.089 4.4.4.4:993 -> 7.7.7.1:53801   .AP.S. 20   4808   9  18412  240
7  01:39:36.697    0.056 4.4.4.4:993 -> 7.7.7.1:53808   .AP..F  3    205  53  29285   68
8  01:39:36.755    0.000 4.4.4.4:993 -> 7.7.7.1:53801   .AP..F  2    141   0      0   70
9  07:18:29.686   27.170 4.4.4.4:993 -> 7.7.7.1:53874   .AP.SF 24   5077   0   1494  211
10 07:18:31.138   25.717 4.4.4.4:993 -> 7.7.7.1:53879   .AP.SF 33  19158   1   5959  580
```

Listing 4.2: Štandardná sieťová prevádzka IMAP serveru (odchádzajúce toky)

```
0  First seen    Duration Src Addr:Port   Dst IP:Port   Flags Pac  Bytes  pps   bps   Bpp
1  12:11:37.824   2.101  6.6.6.1:47785 -> 4.4.4.4:993  .APRSF 17   1419    8   5403   83
2  12:11:37.824   4.632  6.6.6.1:47788 -> 4.4.4.4:993  .APRSF 16   1395    3   2409   87
3  12:11:37.824   2.109  6.6.6.1:47786 -> 4.4.4.4:993  .APRSF 17   1419    8   5382   83
4  12:11:37.824   7.174  6.6.6.1:47778 -> 4.4.4.4:993  .APRSF 17   1439    2   1604   84
5  12:11:37.824   7.160  6.6.6.1:47782 -> 4.4.4.4:993  .APRSF 17   1455    2   1625   85
6  12:11:37.824   6.130  6.6.6.1:47787 -> 4.4.4.4:993  .APRSF 16   1379    2   1799   86
7  12:11:37.824   7.180  6.6.6.1:47784 -> 4.4.4.4:993  .APRSF 17   1439    2   1603   84
8  12:11:37.824   7.166  6.6.6.1:47779 -> 4.4.4.4:993  .APRSF 17   1439    2   1606   84
9  12:11:37.824   7.171  6.6.6.1:47783 -> 4.4.4.4:993  .APRSF 17   1439    2   1605   84
10 12:11:37.824   7.162  6.6.6.1:47781 -> 4.4.4.4:993  .APRSF 17   1455    2   1625   85
```

Listing 4.3: Útok na IMAP server (prichádzajúce toky)

```
0  First seen    Duration Src Addr:Port   Dst IP:Port   Flags Pac  Bytes  pps   bps   Bpp
1  12:11:37.824  2.101  6.6.6.1:993 -> 4.4.4.4:47785  .AP.SF  14   3437    6  13087   245
2  12:11:37.824  4.632  6.6.6.1:993 -> 4.4.4.4:47788  .AP.S.  13   3689    2   6371   283
3  12:11:37.824  2.109  6.6.6.1:993 -> 4.4.4.4:47786  .AP.SF  14   3437    6  13037   245
4  12:11:38.825  6.165  6.6.6.1:993 -> 4.4.4.4:47779  .AP.S.  13   3385    2   4392   260
5  12:11:38.829  6.155  6.6.6.1:993 -> 4.4.4.4:47782  .AP.S.  14   3437    2   4467   245
6  12:11:38.829  6.166  6.6.6.1:993 -> 4.4.4.4:47783  .AP.S.  14   3437    2   4459   245
7  12:11:38.828  6.170  6.6.6.1:993 -> 4.4.4.4:47778  .AP.S.  14   3437    2   4456   245
8  12:11:38.829  6.157  6.6.6.1:993 -> 4.4.4.4:47781  .AP.S.  14   3437    2   4465   245
9  12:11:37.824  6.130  6.6.6.1:993 -> 4.4.4.4:47787  .AP.S.  14   3437    2   4485   245
10 12:11:38.829  6.175  6.6.6.1:993 -> 4.4.4.4:47784  .AP.S.  14   3437    2   4452   245
```

Listing 4.4: Útok na IMAP server (odchádzajúce toky)

Z parameteru **First seen**, ktorý zobrazuje čas príchodu prvého paketu v danom toku, je vidieť, že v prípade štandardnej sieťovej prevádzky môžu byť toky od seba vzdialené niekoľko minút až hodín. V prípade útoku všetky toky od útočníka začínajú v rovnakom čase, z čoho sa dá vyvodiť, že útok bol prevádzaný cez viacero súčasných TCP spojení. Ak by útočník použil len jedno spojenie, budú toky začínať krátko po sebe, avšak počet tokov za jednotku času (napr. 5 minút) bude stále vysoký.

Medzi **dĺžkami trvania tokov** (Duration) v prípade útoku prevláda v našom prípade hodnota 6 až 7 sekúnd v oboch smeroch. Takto dlhý čas jedného pokusu o prihlásenie je spôsobený penalizáciou, keď sa server snaží spomaliť útočníka v hádaní hesiel a v prípade zadania neplatného hesla posiela odpoveď až po určitom čase. Implementácia oneskorenia odpovede nie je u všetkých protokolov štandardom. Aj keď často používané poštové servery Dovecot a Postfix majú túto ochranu povolenú v základnom nastavení, Microsoft Exchange Server odpovede štandardne neoneskoruje. Test útoku na Microsoft Exchange Server je v prílohe B.

IP adresa (Src Addr, Dst IP) útočníka a poštového serveru je v našom prípade stále rovnaká. Tento prípad útoku sa vyskytuje najčastejšie (v našom prípade všetky zachytené útoky). Menej časté sú útoky keď útočník útočí z rôznych IP adries a prípadne aj na viacero poštových serverov súčasne. Port útočníka je obecne náhodný a väčší ako 1000.

TCP príznaky (Flags) sú často rovnaké v danom smere u všetkých tokov patriacich k útoku. Každý pokus o prihlásenie začína paketom s príznakom SYN a končí príznakom FIN. Narozdiel od štandardnej prevádzky, pakety od útočníka často obsahujú príznak RST. Nie je to však pravidlom, čo dokumentuje príklad útoku na Microsoft Exchange server v prílohe B.

Počet paketov v toku (Pac), **počet bajtov** (Bytes) a **priemerný počet bajtov na paket** (Bpp) pri útoku sú konštantné, alebo sa líšia len v rádoch jednotiek, narozdiel od štandardnej prevádzky, kde majú tieto parametre veľký rozptyl. Bežný užívateľ okrem

prihlasovania zobrazuje obsah poštových schránok, sťahuje a upravuje správy, čo má za následok rôzny počet paketov a odlišnú veľkosť. Útočník, ktorý skúša heslá, posiela stále rovnaké dotazy a server mu posiela negatívne odpovede, ktoré majú konštantnú veľkosť. Prvý a posledný tok môžu byť odlišné nezávisle na úspechu/neúspechu útoku.

Počet paketov za sekundu (pps) a **počet bitov za sekundu** (bps) v tokoch útoku sa od štandardnej prevádzky líši podobne ako predchádzajúce tri parametre, ale naviac sa mení podľa záťaže sietí a serverov útočníka a obete.

Z predstavených parametrov vyplýva, že na detekciu útoku je najvhodnejší počet paketov v toku a priemerný počet bajtov na paket spolu s frekvenciou príchodov tokov. Na tieto charakteristiky tokov musíme aplikovať takú štatistickú funkciu, aby jej výsledok jednoznačne rozlíšil útoky od legitímnych tokov. Po zvážení niekoľkých funkcií bol za štatistickú hodnotu zvolený rozptyl, pretože najlepšie vyjadruje rozdiely medzi rozmanitou prevádzkou serveru a homogénnymi charakteristikami tokov v prípade útoku. Rozptyl veličín sa počíta podľa vzorca:

$$\sigma^2 = \frac{1}{n}\sum_{i=1}^{n}(x_i - \mathrm{E}(x))^2$$

kde n je počet prvkov, x aktuálny prvok a $\mathrm{E}(x)$ stredná hodnota prvkov x.

Pre overenie zvolených charakteristík boli na testovacie prostredie uskutočnené desiatky útokov pomocou programov Hydra a Medusa. Tabuľky 4.1 a 4.2 zobrazujú výsledky útokov vygenerované programom Hydra na IMAP, SMTP a POP3 server. Prvá tabuľka obsahuje štatistiky tokov, ktoré smerovali k serverom, druhá tabuľka obsahuje toky v opačnom smere. Pre porovnanie, tabuľka 4.3 obsahuje štatistiky pre útoky vygenerované programom Medusa. Na protokol SMTP sa mi pomocou programu Medusa nepodarilo uskutočniť útok, pretože server odmietal žiadosti o prihlásenie. Použitý wordlist obsahoval 1000 hesiel, útočilo sa na jeden účet z jednej IP adresy paralelne cez 10 vlákien.

	SMTP		POP3		IMAP	
	útok	bez útoku	útok	bez útoku	útok	bez útoku
počet tokov/min.	53.216	0.077	2.253	0.042	10.399	0.249
počet paketov/min.	910.427	4.052	85.932	8.31	226.231	5.551
rozptyl dĺžky tokov	60.663	1.325	1582.798	314.339	63.986	37.220
rozptyl počtu bajtov	123217	3.5×10^9	530117	1.3×10^9	38885	1.3×10^6
rozptyl počtu paketov	10.722	1766.262	49.765	747943	3.058	133.796
rozptyl bajtov/paket	0.46235	139533	2.387	11.685	22.643	218.432

Tabuľka 4.1: Charakteristiky tokov Hydra - (smer k serveru)

	SMTP		POP3		IMAP	
	útok	bez útoku	útok	bez útoku	útok	bez útoku
počet tokov/min.	53.2	0.1	2.2	0.1	10.3	0.2
počet paketov/min.	910.4	2.8	125.2	2.9	237.1	3.5
rozptyl dĺžky tokov	60.6	1.3	1582.8	313.7	57.5	74.6
rozptyl počtu bajtov	235985	800348	1701686	1.7×10^{14}	134597	4.9×10^6
rozptyl počtu paketov	28.3	489.8	146.6	86353	5.5	66.4
rozptyl bajtov/paket	121.6	130.3	57.1	120657	22.6	3910.3

Tabuľka 4.2: Charakteristiky tokov Hydra - (smer od serveru)

Štatistiky v stĺpci 'bez útoku' boli vypočítané z dát reálnej prevádzky. Odchytené boli v prostredí popísanom v sekcii 4.5. Výsledky sú pre konkrétnu IP adresu za časový interval 24 hodín, alebo menej, podľa časového rozsahu komunikácie.

	POP3		IMAP	
	k serveru	od serveru	k serveru	od serveru
počet tokov/min.	34.7	34.7	3.2	3.2
počet paketov/min.	349.1	349.1	69.7	99.4
rozptyl dĺžky tokov	0.1	0.1	0.0	0.0
rozptyl počtu bajtov	422.8	506.8	3101.7	9065.9
rozptyl počtu paketov	0.1	0.0	0.3	0.7
rozptyl bajtov/paket	1.2	2.4	0.0	0.2

Tabuľka 4.3: Charakteristiky tokov - Medusa (oba smery)

Z nazbieraných dát môžeme odvodiť charakteristiky, ktorými sa útoky odlišujú od štandardnej prevádzky.

- veľký počet tokov a paketov za určitý časový interval
- malý rozptyl počtu paketov a bajtov v tokoch patriacich konkrétnemu útoku
- malý rozptyl priemerného počtu bajtov na paket
- prvý a posledný tok sa môžu líšiť od ostatných tokov
- dĺžka tokov je rôzna a závisí od penalizácie a počtu pokusov o prihlásenie v jednom toku

Tieto charakteristiky sú spoločné pre všetky skúmané poštové protokoly. Veľký počet tokov a vyššia frekvencia ich príchodov sú zhodné so štatistikami v článku [9], ktorý sa zaoberá slovníkovými útokmi na protokol SSH. Ostatné parametre - dĺžka tokov a veľkosť paketov sa líšia a závisia hlavne na použitej penalizácii v prípade neuhádnutia hesla. Ak útočník po každom pokuse zmení TCP port, toky majú najčastejšie dĺžku od 0.01 bez penalizácie až po 6 sekúnd v prípade použitia penalizácie. Ak útočník posiela všetky pokusy o prihlásenie s rovnakým zdrojovým portom, toky budú mať dĺžku zodpovedajúcu maximálnemu aktívnemu času timeout - napr. 180 sekúnd.

4.5 Slovníkové útoky v reálnom prostredí

Charakteristiky tokov boli zatiaľ skúmané vo virtuálnom prostredí. Táto podkapitola sa zaoberá tokmi dát a slovníkovými útokmi, ktoré boli zachytené v reálnom prostredí. Cieľom je zistiť, či charakteristiky ako veľký počet paketov a malý rozptyl platia aj pre útoky generované reálnymi útočníkmi, alebo sú tieto útoky sofistikovanejšie.

Na skúmanie reálnych útokov použijem dataset získaný v reálnom prostredí . Dáta boli zbierané a klasifikované od 18.12.2014. Na klasifikáciu boli použité log súbory[10] z poštového serveru, z ktorých bolo možné získať nasledujúce informácie:

- čas prvého pokusu o prihlásenie

[10] spisovne záznamy, ale pre lepšiu čitateľnosť budem používať anglický termín log

- čas posledného pokusu o prihlásenie
- počet pokusov o prihlásenie
- IP adresa útočníka a poštového serveru
- poštový protokol
- skúšané prihlasovacie meno
- pre väčšinu útokov: čas všetkých pokusov o prihlásenie

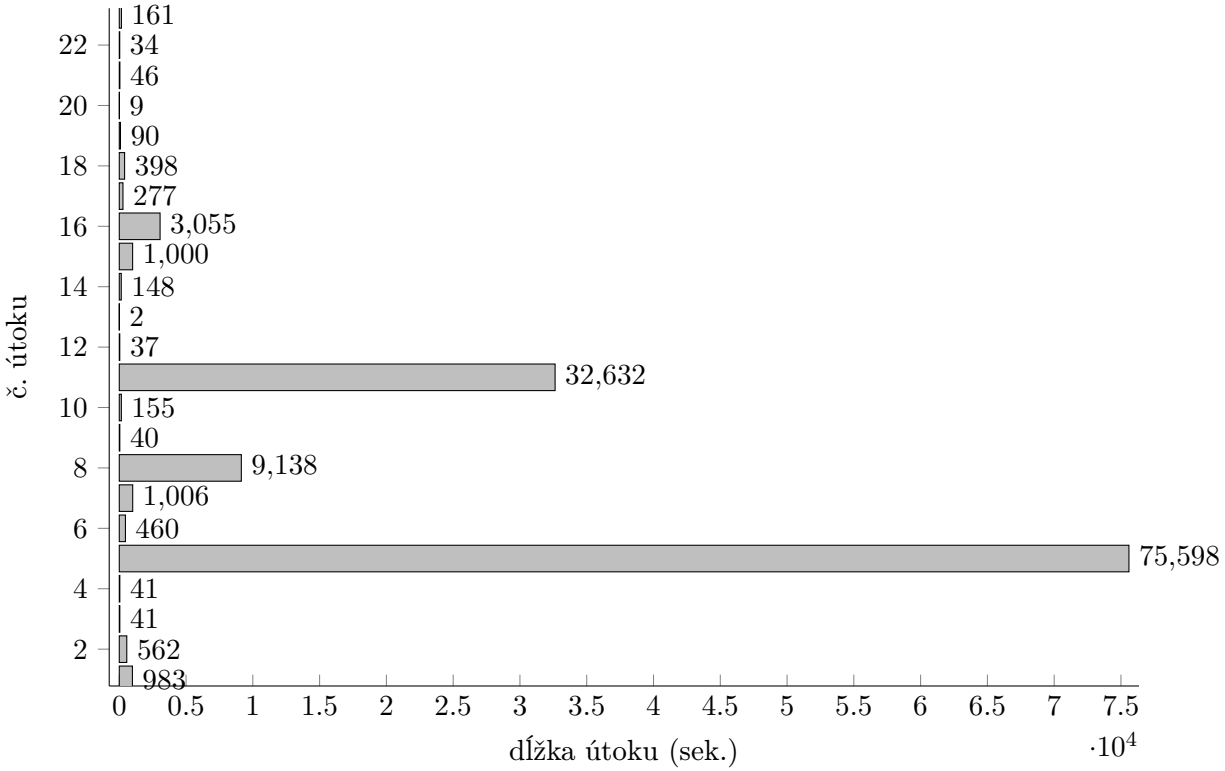
Výpis z poštového serveru 4.5 zobrazuje ukážku útoku na POP3 server v log súboroch. V konkrétnom prípade útočník urobil 18 pokusov o prihlásenie a skúšal stále rovnaké prihlasovacie meno. Jednotlivé pokusy o prihlásenie sú od seba vzdialené jednotky sekúnd a útočník ukončil útok po 18 pokusoch. Toky, ktoré patria k tomuto útoku zobrazuje v smere k serveru výpis 4.6 a v smere od serveru 4.7. Z výpisov je vidieť hlavný rozdiel medzi týmto útokom a útokmi generovanými vo virtuálnom prostredí - útočník použil na každý pokus o prihlásenie odlišný port. Ostatné charakteristiky - dĺžka tokov, rozptyl počtu paketov a bajtov na paket zodpovedajú charakteristikám vo virtuálnom prostredí. Dĺžky trvania útokov pre protokol POP3 zobrazuje graf 4.1, z ktorého je vidieť, že väčšina slovníkových útokov trvá veľmi krátko. Z 23 skúmaných útokov trvalo 13 útokov kratšie ako 5 minút, 10 útokov trvalo kratšie ako 1 minútu. Výnimku tvorí útok, ktorý trval 21 hodín. Počty pokusov o prihlásenie pre jednotlivé útoky zobrazuje graf 4.2. Charakteristiky všetkých útokov pre prichádzajúce toky zobrazuje tabuľka 5.3 v kapitole 5.5. Odchádzajúce toky sú v tabuľke 5.4.

Častou praktikou útočníkov je skúšanie rôznych prihlasovacích mien - útočník má slovník mien a ku každému z nich skúša jedno heslo. Tento typ útoku nezachytia jednoduché blokovania na strane serveru, pretože ku každému účtu sa útočník hlási len raz. Z pohľadu tokov je tento útok zhodný s útokmi na jeden účet. Sofistikovanejšie útoky používajú zoznamy hesiel podľa krajiny, kde sa server nachádza alebo namiesto najčastejších prihlasovacích mien používajú poštové adresy, ktoré na serveri existujú.

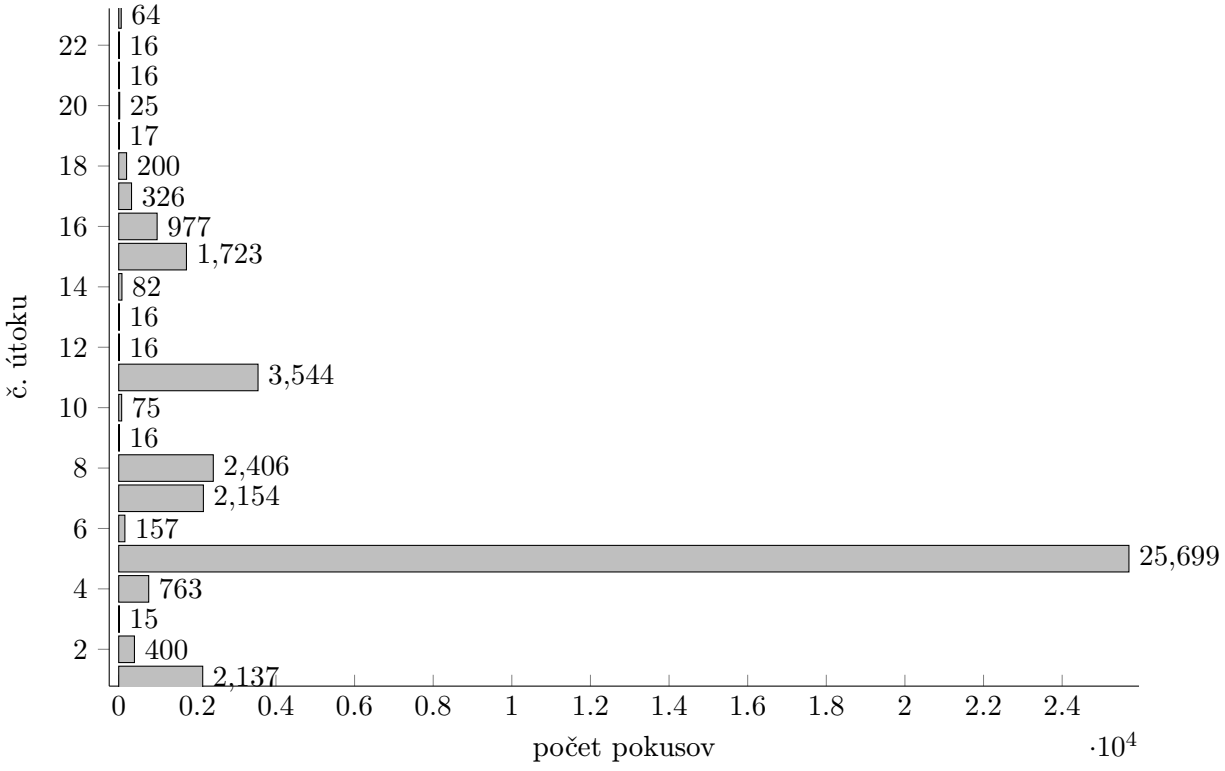
Na poštový server bolo k času písania tohto textu uskutočnených 27 slovníkových útokov, čo zodpovedá približne jednému útoku za 3 dni. Zahrnuté sú len protokoly IMAP a POP3, pretože log súbory SMTP serveru mi neboli prístupné.

```
1 Dec 28 03:45:59 <mail.info> dovecot: pop3-login: Aborted login (auth failed, 1
  attempts): user=<oracle>, method=PLAIN, rip=6.6.6.1, lip=4.4.4.4
2 Dec 28 03:46:05 <mail.info> last message repeated 2 times
3 Dec 28 03:46:06 <mail.info> dovecot: pop3-login: Aborted login (auth failed, 1
  attempts): user=<oracle>, method=PLAIN, rip=6.6.6.1, lip=4.4.4.4
  ...
16 Dec 28 03:46:42 <mail.info> dovecot: pop3-login: Aborted login (auth failed, 1
  attempts): user=<oracle>, method=PLAIN, rip=6.6.6.1, lip=4.4.4.4
17 Dec 28 03:46:44 <mail.info> dovecot: pop3-login: Aborted login (auth failed, 1
  attempts): user=<oracle>, method=PLAIN, rip=6.6.6.1, lip=4.4.4.4
18 Dec 28 03:46:47 <mail.info> dovecot: pop3-login: Aborted login (auth failed, 1
  attempts): user=<oracle>, method=PLAIN, rip=6.6.6.1, lip=4.4.4.4
```

Listing 4.5: Výpis z poštového serveru



Obr. 4.1: Dĺžky jednotlivých útokov na POP3 server



Obr. 4.2: Počet pokusov o prihlásenie pri útokoch na POP3 server

```
First seen  Duration Src Addr:Port Dst Addr:Port Flags  Pac Bytes pps  bps Bpp
03:45:58.375 0.571   6.6.6.1:27501 -> 4.4.4.4:110 .A..S.  3   128   5  1793  42
03:45:58.651 7.872   6.6.6.1:27513 -> 4.4.4.4:110 .APRSF 7   314   0   319  44
03:46:01.531 7.955   6.6.6.1:27634 -> 4.4.4.4:110 .APRSF 7   320   0   321  45
03:46:04.513 7.930   6.6.6.1:27757 -> 4.4.4.4:110 .APRSF 7   322   0   324  46
...
03:46:32.339 7.297   6.6.6.1:28927 -> 4.4.4.4:110 .APRSF 7   320   0   350  45
03:46:39.570 7.858   6.6.6.1:29234 -> 4.4.4.4:110 .APRSF 7   319   0   324  45
03:46:37.262 7.303   6.6.6.1:29135 -> 4.4.4.4:110 .APRSF 7   318   0   348  45
03:48:59.048 0.000   6.6.6.1:27501 -> 4.4.4.4:110 ...R..  1    40   0     0  40
```

Listing 4.6: Toky útoku - smer k serveru

```
First seen  Duration Src Addr:Port Dst Addr:Port Flags  Pac Bytes pps  bps Bpp
03:45:58.375 0.274 4.4.4.4:110   -> 6.6.6.1:27501 .AP.S.  2   108   7  3153  54
03:45:58.651 7.690 4.4.4.4:110   -> 6.6.6.1:27513 .AP.SF  9   439   1   456  48
03:46:01.532 7.776 4.4.4.4:110   -> 6.6.6.1:27634 .AP.SF  9   439   1   451  48
03:46:04.513 7.575 4.4.4.4:110   -> 6.6.6.1:27757 .AP.SF  9   439   1   463  48
...
03:46:34.642 7.442 4.4.4.4:110   -> 6.6.6.1:29022 .AP.SF  9   439   1   471  48
03:46:39.570 7.602 4.4.4.4:110   -> 6.6.6.1:29234 .AP.SF  9   439   1   461  48
03:46:37.262 7.102 4.4.4.4:110   -> 6.6.6.1:29135 .AP.SF  9   439   1   494  48
03:48:58.648 0.000 4.4.4.4:110   -> 6.6.6.1:27501 .A...F  1    40   0     0  40
```

Listing 4.7: Toky útoku - smer od serveru

Útoky zachytené na poštovom serveri spĺňajú charakteristiky útokov, ktoré boli vytvorené na základe dát z virtuálneho prostredia. Reálne prostredie naviac umožnilo sledovať správanie útočníkov, ktoré sa dá zhrnúť do nasledujúcich znakov:

- slovníkové útoky sú krátke - trvajú obvykle desiatky sekúnd
- útočník mení port po každom pokuse o prihlásenie
- pokusy o prihlásenie prichádzajú často s frekvenciou približne 1 pokus za 2 sekundy ale nájdu sa aj útoky, kde sú pokusy od seba vzdialené viac ako 60 sekúnd
- útoky prichádzajú len z jednej IP adresy
- IP adresa útočníka je zo zahraničia

Kapitola 5

Metódy detekcie a implementácia

Z kapitoly 4.4 vyplývajú funkcie, ktoré musí implementovať skript, aby bolo možné detekovať slovníkové útoky:

- detekcia zvýšenej aktivity (veľkého množstva tokov v porovnaní s bežnou prevádzkou)
- výpočet rozptylu počtu paketov a bajtov na paket
- výpočet hraničných hodnôt pre detekciu útoku
- výpočet rozptylu pre spätné toky
- špecifikácia parametrov zvlášť pre SMTP, POP3 a IMAP

5.1 Algoritmus detekcie útokov

Vstupom algoritmu je slovník tokov, ktorého kľučom je zdrojová IP adresa a hraničné hodnoty parametrov na detekciu útokov. Výstup je zoznam IP adries, z ktorých boli detekované útoky a parametre útokov: cieľová IP adresa, čas začiatku útoku, dĺžka trvania, počet tokov a rozptyl paketov.

Pre každú unikátnu IP adresu prebieha nasledujúci proces detekcie útoku. Najskôr sa algoritmus snaží nájsť postupnosť tokov, ktorá je dlhšia ako stanovená hranica. V postupnosti musia byť susedné toky vzdialené menej ako zvolený časový interval. Týmto spôsobom môžeme určiť, že útoky budeme detekovať len v prípadoch, keď sme z danej IP adresy dostali viac ako n tokov a tie neboli od seba vzdialené viac ako x sekúnd. Vylúčime tým veľmi krátke a časovo vzdialené pokusy.

Ak algoritmus detekuje postupnosť tokov, ktorá spĺňa vyššie uvedené parametre, skontroluje, či rozptyl počtu paketov a bajtov na paket vo všetkých tokoch pre danú IP adresu je menší ako definovaná hranica. Ak je rozptyl menší, znamená to, že užívateľ opakoval rovnakú akciu viackrát za sebou a môže ísť o útok. Algoritmus porovná rozptyly parametrov spätných tokov pre danú IP adresu s definovanými hranicami a ak sú vypočítané rozptyly menšie, IP adresa je označená ako IP adresa útočníka.

V pôvodnej verzii algoritmu sa počítali rozptyly parametrov len pre nájdenú podmnožinu tokov a nie pre všetky toky danej IP adresy. Pri testovaní sa ale zistilo, že tento prístup spôsobuje veľa falošných pozitívnych detekcií, keďže podmnožina tokov, ktorá charakteristikami zodpovedá útoku sa dá často nájsť aj v sieťovej prevádzke, ktorá neobsahuje útoky.

Výpočet charakteristík pre všetky toky vychádza z predpokladu, že legitímny užívateľ vykonáva, alebo v minulosti vykonal operácie, ktoré generujú heterogénne charakteristiky tokov. Aktuálne toky, ktoré majú veľkú frekvenciu a často malý rozptyl charakteristík (napr. odoslanie veľkého počtu mailov) sa analyzujú spolu so staršími tokmi pre danú IP adresu a tým sa zabraňuje falošným detekciám útokov.

Výpis 5.1 zobrazuje pseudokód algoritmu.

```
for IP in flowDictionary:      # pre každú IP adresu zo slovníka
  for flow in IP:              # pre každý tok IP adresy

    ''' počítaj toky '''
    if (startTime - lastStartTime) < maxInterval:
      counter += 1
      continue

    ''' interval medzi tokmi bol prekročený, ale nemáme dostatok
        tokov '''
    if counter < flowLimit:
      discard_previous_flows()
      continue

    ''' možný útok - skontroluj rozptyly '''
    if isAttack(IP):

    if isAttack(IPreverse): # detekovaný útok,skontroluj opačné toky
        print "Detekovaný útok"
```

Listing 5.1: Presudokód algoritmu detekcie útokov

V rámci tejto práce boli implementované dva skripty - skript na detekciu útokov, ktorý označujem názvom *dictatt* a ďalší na zistenie hraničných hodnôt s názvom *learnchar*.

5.2 Skript - detekcia útokov

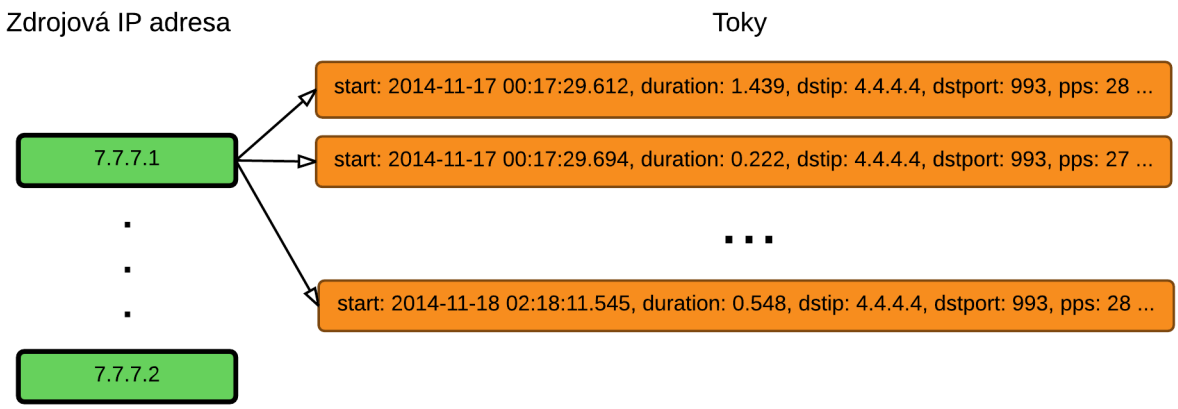
Skript bol napísaný v jazyku Python. Povinné parametre sú cesta k nfdump súborom a názov exportéru. Voliteľne je možné definovať časový interval v ktorom sa budú hľadať útoky, IP adresu a meno používateľa na serveri kde beží nfdump a protokol (SMTP, POP3, IMAP).

Parametre, ktoré ovplyvňujú citlivosť detekcie sa načítajú z konfiguračného súboru. Minimálny počet a frekvencia príchodu tokov sú spoločné pre všetky protokoly. Minimálny rozptyl paketov a bajtov na paket sú definované pre každý protokol zvlášť. Tieto parametre sa definujú oddelene aj pre jednotlivé smery tokov — *k* a *od* serveru. Konfiguračný súbor obsahuje celkovo 14 parametrov — 2 spoločné a 4 pre každý protokol.

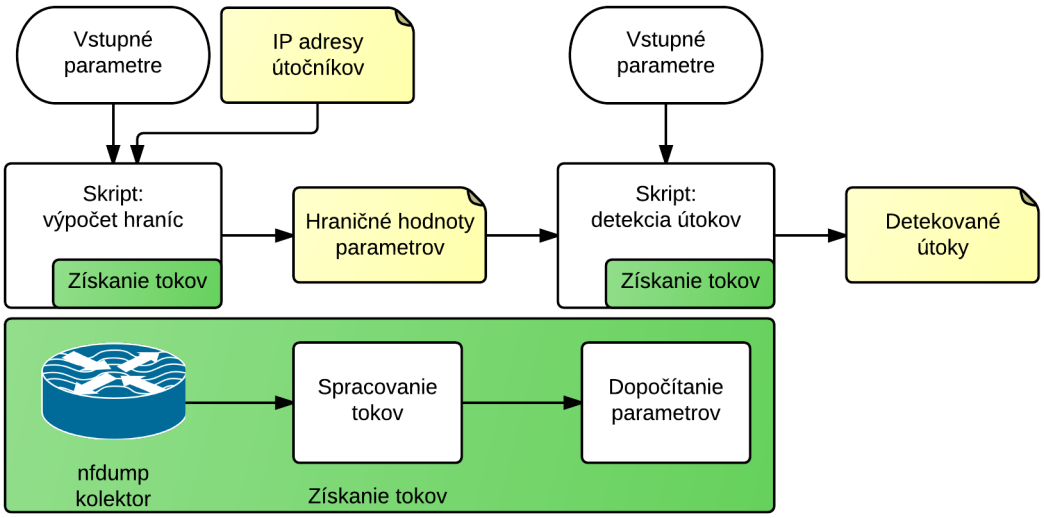
Aby administrátor nemusel hľadať vhodné hraničné hodnoty parametrov pre konkrétnu sieť, napísal som skript, ktorý tieto hranice vypočíta z tokov a znalosti IP adries útočníkov. Skript je popísaný v kapitole 5.3, táto sekcia sa zaoberá len detekciou útokov. Prepojenie a vstupy a výstupy skriptov znázorňuje diagram 5.2.

Skript posiela dotazy kolektoru nfdump. Nfdump môže byť inštalovaný lokálne, alebo na vzdialenom serveri, z ktorého sa toky stiahnu po prihlásení pomocou SSH. Pre každý poštový protokol sa pošlú dva dotazy, ktoré obsahujú príkaz pre nfdump. Prvý dotaz je pre toky smerujúce k serveru, ďalší pre opačný smer. Server odošle skriptu zoznam tokov (výstup z nfdump). Jednotlivé toky sú ukladané do slovníka, kde kľúčom je v prvom prípade zdrojová

IP adresa. V druhom prípade je kľúč cieľová IP adresa, ktorá zodpovedá zdrojovej IP v prvom slovníku. Štruktúra slovníka je na obrázku 5.1. Zároveň s pridaním toku do slovníka sa dopočítajú parametre, ktoré bude skript potrebovať k analýze tokov: dĺžka trvania toku, počet paketov za sekundu a počet bajtov na paket. Algoritmus detekcie vyhodnotí toky pre každú IP adresu a ak nájde útok, vypíše jeho štatistiky na štandardný výstup.



Obr. 5.1: Štruktúra uloženia tokov



Obr. 5.2: Vstupy/výstupy a prepojenie skriptov

Príklad výstupu skriptu sa nachádza tu 5.4.

5.3 Skript - výpočet hraničných hodnôt

Tento skript pomáha nastaviť hraničné hodnoty vstupných parametrov pre skript na detekciu útokov. Aby bolo možné tieto hodnoty vypočítať pre konkrétny server, administrátor serveru musí zadať do konfiguračného súboru niekoľko IP adries, z ktorých sa na daný server útočilo. Skript vyhľadá podmnožiny tokov, ktoré zodpovedajú útokom a vypočíta z nich hraničné hodnoty. Výsledné hraničné hodnoty sú maximom získaných hodnôt. Aby bolo možné správne detekovať podmnožiny tokov, musí niektoré parametre detekcie zadať administrátor. Skript na základe týchto parametrov nájde zvyšné hraničné hodnoty.

- Hranice zadané administrátorom:
  - minimálny počet tokov potrebný na detekciu útoku
  - maximálny interval medzi príchodmi tokov
- Detekované hranice (pre každý protokol):
  - rozptyl počtu paketov v toku (v smere k serveru)
  - rozptyl počtu bajtov na paket (v smere k serveru)
  - rozptyl počtu paketov v toku (v smere od serveru)
  - rozptyl počtu bajtov na paket (v smere od serveru)

Výsledné hranice nie sú ideálne hodnoty, ktoré by mal administrátor použiť bezo zmeny, ale maximálne hodnoty rozptylu vypočítané z daných útokov. Ich presnosť závisí na počte poskytnutých IP adries útočníkov. Pre detekčný algoritmus to znamená že je schopný detekovať útoky s najvyšším rozptylom takým, aký mal najzložitejší útok zo zoznamu daných IP adries, čo je veľmi obmedzujúce. Preto odporúčam mierne zvýšiť hodnoty, ktoré skript vypočítal.

5.4 Ukážka použitia a výstupu algoritmov

Ako ukážku možností implementovaných nástrojov uvádzam príklad použitia skriptov. Vstupom je dataset č. 2 popísaný v kapitole 5.5.2 a log z poštového serveru, ktorý obsahuje IP adresy útočníkov. Ak by sme nemali informácie o útočníkoch, je možné vygenerovať útoky na náš server pomocou skriptov (hydra, medusa alebo iné). Na generovanie útokov je nutné použiť IP adresu, z ktorej sa nepristupovalo na poštový server, alebo obmedziť časové hranice v algoritme tak, aby do charakteristík tokov neboli zahrnuté legitímne prístupy z danej IP. Ak útoky z nejakých dôvodov nemôžeme vygenerovať (nejedná sa o testovací server) môžeme zvoliť hodnoty ručne a následne podľa výstupu skriptu ich upraviť tak, aby bolo množstvo nájdených útokov čo najvyššie a výstup neobsahoval falošné detekcie.

Príklad vstupného súboru attacks.conf zobrazuje výpis 5.2. IP adresy sú anonymizované.

```
smtp: 6.6.6.1, 6.6.6.2, 6.6.6.3
imap: 6.6.6.4, 6.6.6.2, 6.6.6.6
pop3: 6.6.6.8, 6.6.6.3, 6.6.6.22
```

Listing 5.2: Vstupný súbor attacks.conf

Po vložení IP adries do konfiguračného súboru bol spustený skript *learnchar*, ktorý vypočítal hraničné hodnoty rozptylov a uložil ich do súboru params.conf. Vypočítané rozptyly boli vynásobené číslom 1.5 a v prípade veľmi malej hodnoty navýšené, aby sme umožnili detekciu zložitejších útokov. Výsledný súbor zobrazuje výpis 5.3.

```
[sensitivity]
minatt = 10              - minimálny počet tokov pre detekciu útoku
frequency = 1            - maximálny interval medzi príchodmi tokov (v min.)
smtp_bppvar_in = 5       - hodnoty pre smtp
smtp_pacvar_out = 3
smtp_bppvar_out = 10
smtp_pacvar_in = 3
imap_bppvar_in = 24      - hodnoty pre imap
imap_pacvar_out = 3
imap_bppvar_out = 113
imap_pacvar_in = 3
pop3_bppvar_in = 41      - hodnoty pre pop3
pop3_pacvar_out = 5
pop3_bppvar_out = 30
pop3_pacvar_in = 5

bppvar - maximálny rozptyl počtu bajtov na paket
pacvar - maximálny rozptyl počtu paketov v tokoch
in/out - prichádzajúce/odchádzajúce toky vzhľadom k poštovému serveru
```

Listing 5.3: Upravený výstup skriptu na detekciu hraničných hodnôt

Súbor params.conf bol okopírovaný do adresára, kde sa nachádza detekčný algoritmus, ktorý po spustení vygeneroval zoznam útokov. Ukážka výstupu skriptu (1 útok) zobrazuje výpis 5.4.

```
Protocol pop3 :
------------------------------------------------------------
| Attacker IP address:      6.6.6.50
| Mail server IP address and port:  4.4.4.4 : 110
| Attack duration: from  2015-01-20 01:25:19 to 2015-01-20 01:27:43
| Attack duration: from  2015-01-23 11:34:10 to 2015-01-23 11:40:12
| IP time window:  2015-01-20 01:25:19 --> 2015-01-23 11:40:12
| -> Flows TO server:
|   Number of flows:   75
|   Flows/minute:      29.849
|   Packets/minute:    208.944
|   Packets variance:  0.0
|   Bytes/packet var:  0.382
| <- Flows FROM server:
|   Number of flows:   75
|   Flows/minute:      29.864
|   Packets/minute:    240.502
|   Packets variance:  0.052
|   Bytes/packet var:  0.164
------------------------------------------------------------
```

Listing 5.4: Ukážka výstupu detekčného skriptu

Z výstupu je vidieť IP adresa útočníka a poštového serveru, časy trvania útokov a časové okno, z ktorého sa počítali štatistiky pre toky. Štatistiky - počet tokov, počet tokov za minútu, počet paketov za minútu, rozptyl paketov a bajtov na paket sú zobrazené zvlášť pre toky smerujúce k serveru a od serveru.

Ak sa použije prepínač *–graphs*, skript vygeneruje histogramy počtu paketov a bajtov na paket pre jednotlivé IP a uloží ich vo formáte pdf do súboru report.pdf. Generovanie

histogramov je časovo veľmi náročné, preto je počet vygenerovaných histogramov obmedzený. Generujú sa histogramy pre IP adresy útočníkov a maximálne 10 histogramov pre IP adresy legitímnych používateľov (pre porovnanie charakteristík). Ukážka histogramov je v prílohe C.

5.5 Testovacie datasety

Skript na detekciu útokov bol otestovaný na viacerých datasetoch. Jeden dataset bol vytvorený vo virtuálnom prostredí, ďalšie dva datasety pochádzajú z produkčných prostredí. Zatiaľ čo vo virtuálnom prostredí nebol problém s koreláciou dát, pretože som vedel presný čas začiatku útokov, počet pokusov a IP adresy útočníka, v dátach z reálnych prostredí museli byť tieto informácie vyhľadávané v log súboroch z poštových serverov. V čase písania tejto práce som nemal dostupných viac datasetov pretože log súbory z poštových serverov obsahujú citlivé dáta, ktoré nikto voľne neposkytuje.

5.5.1 Dataset č.1

Dataset obsahuje spolu 24 útokov na servery SMTP, IMAP a POP3, ktoré boli nainštalované vo virtuálnom prostredí predstavenom v kapitole 4.3.1. Na každý z uvedených poštových protokolov bolo uskutočnených 8 útokov. Prvé 3 útoky boli generované programom Medusa, 5 útokov bolo generovaných pomocou programu Hydra. Počet pokusov, vlákien a čas trvania útokov bol zhodný pre všetky 3 poštové protokoly a tieto parametre zobrazuje tabuľka 5.1.

útok	program	počet vlákien	dĺžka wordlistu	dĺžka útoku	interval medzi pok.
1	medusa	10	50	78 s	0 s
2	medusa	10	250	417 s	0 s
3	medusa	1	50	504 s	0 s
4	hydra	10	50	95 s	0 s
5	hydra	10	250	621 s	0 s
6	hydra	1	50	314 s	0 s
7	hydra	1	50	727 s	10 s
8	hydra	1	50	936 s	30 s

Tabuľka 5.1: Útoky v datasete č. 1

Dataset obsahuje sekvenčné aj paralelné útoky realizované cez 10 vlákien súčasne. IP adresa útočníka sa počas útoku nemenila. Pri posledných dvoch útokoch bol vložený medzi jednotlivé pokusy časový interval 10 resp. 30 sekúnd. Pri 30 sekundách ale IMAP server ukončoval spojenia z dôvodu nečinnosti, ktoré Hydra okamžite naväzovala a posielala ďalší pokus, preto je výsledný čas útoku kratší ako predpokladaný.

Okrem útokov obsahuje tento dataset toky, ktoré obsahujú legitímnu prevádzku poštového serveru, ktorá by mohla byť detekovaná ako útok: odoslanie a prijatie veľkého počtu mailov naraz a periodické kontrolovanie poštovej schránky.

5.5.2 Dataset č.2

Tento dataset obsahuje toky zachytené v produkčnom prostredí 4.5 od 18.12.2014 do 18.3.2015. Počas tejto doby bolo uskutočnených na sledovaný poštový server 27 slovníkových útokov.

Útoky boli detekované a klasifikované na základe výpisov z poštového serveru, ktorý zaznamenával všetky pokusy o prihlásenie na služby IMAP a POP3. Log súbory SMTP serveru mi neboli prístupné. Dataset obsahuje 12.5 milióna tokov, dátová prevádzka mailserveru za daný časový interval bola 107 GB. Tabuľka 5.2 zobrazuje počty útokov na jednotlivé protokoly, minimálnu, maximálnu a priemernú dĺžku útoku v sekundách, frekvenciu a počty pokusov o prihlásenie. Tabuľky 5.3 a 5.4 zobrazujú parametre útokov tak, ako ich zachytil kolektor nfdump. Útokom v tomto datasete sa podrobne venovala kapitola 4.5.

	IMAP				POP3			
	min.	max.	priemer	medián	min.	max.	priemer	medián
počet útokov	4				23			
dĺžka útoku (sek.)	1	233	80.3	17	2	75598	5474.4	161
počet pokusov	10	418	149	19	15	25699	1775.8	157
frekvencia (pokusov/sek.)	0.5	10	4.2	2.2	0.1	16.6	1.8	0.5

Tabuľka 5.2: Útoky v datasete č. 2

Na POP3 bolo počas troch mesiacov uskutočnených 23 útokov (v porovnaní len so 4 útokmi na IMAP).

	IMAP				POP3			
	min.	max.	priemer	medián	min.	max.	priemer	medián
počet tokov/min.	0.003	201.3	78.8	56.9	9.8	127.2	53.2	31.3
počet paketov/min.	0.003	2617.4	797.2	285.7	62.2	893.1	358.1	213.7
rozptyl počtu paketov	0	0.937	0.355	0.243	0.082	5.8	1.8	0.8
rozptyl bajtov/paket	0	23.4	6.4	1.2	0.346	4.7	1.9	0.6

Tabuľka 5.3: Dataset č. 2 - parametre prichádzajúcich tokov

	IMAP				POP3			
	min.	max.	priemer	medián	min.	max.	priemer	medián
počet tokov/min.	0.003	194.6	77.1	56.9	9.2	126.7	59.5	29.1
počet paketov/min.	0.019	2160.2	709.5	338.9	80.7	1139.8	509.4	254.6
rozptyl počtu paketov	0.09	0.97	0.363	0.196	0	11.9	2.44	0.03
rozptyl bajtov/paket	0.475	112.3	38.4	20.5	0	18.5	3.7	0.116

Tabuľka 5.4: Dataset č. 2 - parametre odchádzajúcich tokov

5.5.3 Dataset č.3

Dataset obsahuje toky z hraničného smerovača na FIT VUT v Brne za časový interval 5 minút. Počet tokov je 1.7 milióna, dátová prevádzka 55 GB. Hustota tokov je kvôli veľkosti siete oveľa vyššia ako v datasete č. 2 a zachytávané boli všetky toky na danom porte. Dataset obsahuje 1 útok, ktorý trval 82 sekúnd a útočník spravil dokopy 17 pokusov o prihlásenie - 6 na POP3 (z toho 3 pokusy na port 110 a 3 na 995), 6 na IMAP a 5 pokusov o prihlásenie na SMTP (opäť na rôzne porty).

Kapitola 6

Vyhodnotenie testov

Táto kapitola obsahuje výsledky testovania skriptu na troch datasetoch popísaných v kapitole 5.5. Pre každý dataset sú najskôr pomocou skriptu 5.3 vypočítané hraničné hodnoty vstupných parametrov. Dané hodnoty sú následne použité ako vstup detekčného skriptu a na záver sú zhodnotené výsledky skriptov.

6.1 Testovanie pre dataset č.1

IP adresy, z ktorých prebiehali prvé tri útoky boli použité na zistenie hraničných hodnôt pomocou skriptu *learnchar*. Hraničné hodnoty boli následne vynásobené konštantou 3 a upravené podľa histogramov, ktoré vygeneroval detekčný skript. Výsledky detekcie zobrazuje tabuľka 6.1.

Útok	Program	Počet vlákien/ počet hesiel/ dĺžka útoku	Detekované/ Všetky útoky		
			SMTP	IMAP	POP3
1	medusa	10 / 50 / 78 s	1/1	1/1	1/1
2	medusa	10 / 250 / 417 s	1/1	1/1	1/1
3	medusa	1 / 50 / 504 s	0/1	0/1	1/1
4	hydra	10 / 50 / 95 s	1/1	1/1	1/1
5	hydra	10 / 250 / 621 s	1/1	1/1	1/1
6	hydra	1 / 50 / 314 s	1/1	1/1	0/1
7	hydra	1 / 50 / 727 s	1/1	1/1	0/1
8	hydra	1 / 50 / 936 s	1/1	1/1	0/1

Tabuľka 6.1: Detekcia útokov v datasete č. 1

Skript detekoval 19 útokov z 24 a nezaznamenal žiadnu falošnú pozitívnu detekciu. Všetkých 5 útokov, ktoré neboli detekované malo 2 spoločné vlastnosti - boli príliš krátke (50 pokusov o prihlásenie) a útočník počas útoku nemenil port, z ktorého posielal pokusy o prihlásenie. Útoky z rovnakého portu negenerujú nový tok pri každom pokuse o prihlásenie, ale až po vypršaní aktívneho časovača. Generované toky sú veľmi dlhé a je ich málo - útoky na SMTP a IMAP vygenerovali len 3 toky, útoky na POP3 3 až 6 tokov, čo je pod minimálnou zvolenou hranicou detekcie (10 tokov). Vygenerované toky sa nelíšia od legitímnej prevádzky poštového serveru a nie je možné z aktuálnych výstupov programu nfdump zistiť, či ide o útok. Dôvod, prečo neboli zachytené niektoré útoky pre POP3 a iné pre SMTP a IMAP je v použití rôznych programov - hydra a medusa používajú zásuvné

moduly pre jednotlivé protokoly, ktorých implementácie sa líšia.

6.2 Testovanie pre dataset č.2

Ako vstup pre skript, ktorý hľadá hraničné hodnoty boli použité tri IP adresy pre IMAP a štyri adresy pre POP3 server, z ktorých boli uskutočnené útoky. Nájdené hodnoty som následne upravil, aby som zvýšil pravdepodobnosť detekcie útoku. Pre SMTP som nemal informáciu o útokoch, preto som použil hodnoty pre POP3. Minimálny počet tokov, ktorý je potrebný na detekciu útoku je desať, frekvencia príchodu tokov bola zvolená jeden tok za minútu. Zvolené vstupné hodnoty zobrazuje výpis 6.1. Vysvetlenie parametrov je tu 5.3.

```
Hodnoty nájdené skriptom:

minatt =            10
frequency =          1
smtp_bppvar_in =     0
smtp_pacvar_out =    0
smtp_bppvar_out =    0
smtp_pacvar_in =     0
imap_bppvar_in =    24
imap_pacvar_out =    1
imap_bppvar_out =  113
imap_pacvar_in =     1
pop3_bppvar_in =     5
pop3_pacvar_out =   12
pop3_bppvar_out =   19
pop3_pacvar_in =     6
```

```
Zvolené vstupné hodnoty:

minatt =            10
frequency =          1
smtp_bppvar_in =     8
smtp_pacvar_out =   15
smtp_bppvar_out =   25
smtp_pacvar_in =     8
imap_bppvar_in =    30
imap_pacvar_out =    3
imap_bppvar_out =  130
imap_pacvar_in =     3
pop3_bppvar_in =     8
pop3_pacvar_out =   15
pop3_bppvar_out =   25
pop3_pacvar_in =     8
```

Listing 6.1: Vstupné hraničné hodnoty

Algoritmus detekoval 26 útokov z 27, nedetekoval 1 útok na POP3, preto som pre tento protokol zvýšil hranice detekcie 1.5-násobne. Tabuľka 6.2 ukazuje výsledky po úprave hraničných hodnôt. Algoritmus detekoval všetky útoky bez falošných detekcií. Celkový čas behu skriptu bol 4 minúty a 10 sekúnd. Samotné hľadanie útokov trvalo 46 sekúnd, zostávajúci čas sa čakalo na stiahnutie tokov zo vzdialeného serveru.

Protokol	**POP3**	**IMAP**
Počet útokov v datasete	23	4
Počet detekovaných útokov	23	4
Falošné detekcie	0	0

Tabuľka 6.2: Úspešnosť detekcie - 2. beh, dataset č. 2

6.3 Testovanie pre dataset č.3

Z dôvodu malého počtu útokov v datasete nebol použitý algoritmus na zistenie prahových hodnôt - prahové hodnoty boli zvolené ručne. Útoky v datasete sú krátke, preto bolo nutné zvoliť veľmi nízky minimálny počet pokusov (5) pre detekciu útoku. Pri testovaní sa ale ukázalo, že dĺžka datasetu (5 minút) nie je dostatočná pre vylúčenie falošných detekcií.

Algoritmus nemá možnosť skontrolovať históriu tokov pre IP adresy, ktoré vygenerovali minimálne 5 tokov s vysokou frekvenciou a tieto boli označené za útoky.

6.4 Zhodnotenie výsledkov testov

Z testov na datasetoch vyplýva, že na úspešnosť detekcie má vplyv nastavenie hraničných hodnôt. Skript *learnchar* síce nájde maximálne hodnoty rozptylov pre vložené IP adresy, ale používateľ musí tieto hodnoty vynásobiť vhodnou konštantou, prípadne skontrolovať podľa vygeneronavých histogramov, aby boli detekované všetky útoky. Nastavenie príliš vysokých hraničných hodnôt môže spôsobiť falošné detekcie. Testy skriptov ale ukázali, že interval na nastavenie hodnôt je dostatočne veľký na to, aby falošné detekcie nevznikali.

Po nastavení hraničných hodnôt zachytil algoritmus všetky útoky z reálneho prostredia v datasete č.2 a nezaznamenal žiadnu falošnú detekciu. Napriek tomuto výsledku sa mi v testovacom prostredí podarilo vygenerovať útoky, ktoré nie je možné detekovať z výstupov nfdump, pretože útoky nevygenerovali dostatočný počet tokov. Tento problém nie je chybou detekčného skriptu a nie je ho možné riešiť na úrovni kolektora, pretože sa dané toky nijak nelíšia od legitímnej sieťovej prevádzky. V kapitole 7 sa pokúsim navrhnúť spôsob detekcie takýchto útokov, ak by sonda získavala z paketov viac parametrov.

Zistenie hraničných hodnôt trvalo pre dataset č.2 skriptu *learnchar* 8.15 sekúnd. Vyhľadanie útokov pre protokoly SMTP, IMAP a POP3 trvalo skriptu 47.22 sekúnd. Dataset obsahoval všetky toky poštového serveru nazbierané počas troch mesiacov. Pre dataset č. 3, trvalo zistenie hraničných hodnôt 2.97 sekundy, nájdenie útokov zabralo skriptu 3.89 sekundy. Dataset obsahoval toky za časový interval 5 minút. Testy prebiehali na bežnom notebooku s procesorom Intel Core i5 a SSD diskom. Toky boli uložené lokálne a nebol použitý prepínač *–graphs*. Z výsledkov vyplýva, že časová náročnosť skriptu je nízka a skript je možné použiť na hľadanie útokov vo veľkých datasetoch na bežne dostupnom hardvéri.

6.5 Porovnanie s ostatnými prístupmi

Na detekciu slovníkových útokov existuje niekoľko prístupov. Obecne sa dajú rozdeliť do dvoch kategórií: detekcia/blokovanie útokov na koncových zariadeniach a detekcia/blokovanie na sieťových zariadeniach. Niektoré z prístupov som spomenul v kapitole 4. Táto kapitola porovnáva výhody a nevýhody implementovaných skriptov s existujúcimi riešeniami.

6.5.1 Detekcia a blokovanie útokov na koncových zariadeniach

Detekcia na koncových zariadeniach sa vykonáva najčastejšie pomocou prehľadávania log súborov poštových serverov. Tento spôsob detekcie je pomerne spoľahlivý, pretože server má detailné informácie o frekvencii a počte neúspešných pokusov o prihlásenie. V konfiguračnom súbore je potrebné definovať formát správy, ktorý zodpovedá neúspešnému pokusu o prihlásenie a detekčný skript bude sledovať výskyt týchto správ za určité časové obdobie. Po prekročení stanovených limitov sa IP adresa útočníka vloží do tabuľky blokovaných adries a dočasne sa zablokuje pomocou pravidla firewall-u. Tento spôsob využíva napríklad sada skriptov fail2ban a čiastočne Logwatch - ten sa ale venuje len detekcii útokov.

- Výhody detekcie na koncových zariadeniach:
  - presná detekcia - server má všetky potrebné informácie o útoku

  - existujúce bezplatné nástroje
  - nie je potrebné samostatné zariadenie

- Nevýhody:
  - nutnosť nasadiť nástroje na všetky servery
  - administrátor nemá globálny prehľad o detekovaných a blokovaných útokoch
  - blokovanie útoku len pre lokálny server
  - detekcia zaťažuje koncové zariadenia

6.5.2 Detekcia a blokovanie útokov na sieťových zariadeniach

Na detekciu útokov sa na sieťovej úrovni používajú zariadenia IDS (Intrusion Detection System), na blokovanie útokov IPS (Intrusion Prevention System). Známym voľne dostupným IPS systémom je Snort. Snort analyzuje sieťovú prevádzku v reálnom čase. Môže analyzovať rôzne protokoly a vyhľadávať v nich útoky - pretečenie zásobníkov, skenovanie portov a iné. Nástroj neobsahuje žiadne algoritmy na detekciu slovníkových útokov. Pravidlá pre detekciu a z nich vyplývajúce akcie si musí definovať administrátor sám. Keďže bol pôvodne navrhnutý na vyhľadávanie signatúr na 7. vrstve ISO/OSI, pri rýchlostiach okolo 1Gb/s degraduje výkon siete [9].

Nástroj, ktorý umožňuje na sieťovej úrovni detekovať a blokovať slovníkové útoky na poštové servery sa mi nepodarilo nájsť. Niektoré IDS a IPS umožňujú detekciu slovníkových útokov ako podmnožinu útokov hrubou silou. Slovníkové útoky sú ale narozdiel od útokov hrubou silou oveľa kratšie a majú nižšiu intenzitu. V kapitole 4.2 je predstavených niekoľko existujúcich riešení detekcie slovníkových útokov na SSH, RDP a LDAP servery. Tieto prístupy porovnám s vytvoreným skriptom.

Detekcia útokov na SSH server

Z dvoch prístupov detekcie útokov na SSH server sa pri porovnaní zameriam len na prvý, pretože bol implementovaný a otestovaný v reálnom prostredí. Druhý prístup je skôr teoretický a veľmi odlišný od implementovaného skriptu, preto by bolo porovnanie zbytočné.

- Spoločné vlastnosti implementovaných nástrojov:
  - meranie času medzi príchodmi paketov od útočníka
  - nutnosť nastavenia počiatočných hraníc na detekciu útoku

- Rozdielne vlastnosti nástroja:
  - použitie konkrétnych veľkostí a počtu paketov ako parameter
  - detekcia úspešnosti útoku na základe posledného toku
  - detekcia útokov z viacerých IP adries
  - dynamické hodnoty parametrov - menia sa počas behu

Autori použili ako charakteristiku útoku konkrétnu veľkosť a počet paketov. Tento prístup sa nedá použiť pri detekcii útokov na poštové servery, pretože veľkosti paketov a ich počet sa pre rôzne protokoly (SMTP, IMAP, POP3) líšia. Veľkosti a počty paketov sa líšia aj pre rôzne implementácie daných protokolov - charakteristiky pre IMAP server Dovecot

neplatia pre Microsoft Exchange Server. Z tohto dôvodu považujem rozptyl parametrov za lepšiu metriku ako konkrétne veľkosti paketov. Autori naviac detekujú z rozdielneho posledného toku úspešnosť útoku. Táto charakteristika tokov sa nepotvrdila u poštových protokolov. Útoky v reálnom aj testovacom prostredí mali často odlišný práve prvý a posledný tok aj napriek tomu, že neboli úspešné. Príklad zobrazuje výpis 6.2.

```
first seen     Duration  Proto  Src IP:Port          Dst IP:Port          Pac    Bytes  Bpp
12:28:25.110   14.045    TCP    6.6.6.1:993    ->    4.4.4.4:53406        22     4457   203
12:28:39.154   18.041    TCP    6.6.6.1:993    ->    4.4.4.4:53410        23     4509   196
12:28:57.194   18.050    TCP    6.6.6.1:993    ->    4.4.4.4:53413        23     4509   196
12:29:15.244   18.051    TCP    6.6.6.1:993    ->    4.4.4.4:53415        23     4509   196
12:29:33.294   18.050    TCP    6.6.6.1:993    ->    4.4.4.4:53417        23     4509   196
12:29:51.344   18.039    TCP    6.6.6.1:993    ->    4.4.4.4:53419        23     4509   196
12:30:09.383   18.040    TCP    6.6.6.1:993    ->    4.4.4.4:53421        23     4509   196
12:30:27.423   18.042    TCP    6.6.6.1:993    ->    4.4.4.4:53423        23     4509   196
12:30:45.464   18.041    TCP    6.6.6.1:993    ->    4.4.4.4:53426        23     4509   196
12:31:03.504   17.531    TCP    6.6.6.1:993    ->    4.4.4.4:53428        23     4509   196
12:31:21.035   18.049    TCP    6.6.6.1:993    ->    4.4.4.4:53430        23     4509   196
12:31:39.084   18.030    TCP    6.6.6.1:993    ->    4.4.4.4:53433        23     4509   196
12:31:57.113   18.042    TCP    6.6.6.1:993    ->    4.4.4.4:53435        23     4509   196
12:32:15.154   18.049    TCP    6.6.6.1:993    ->    4.4.4.4:53438        23     4509   196
12:32:33.203   18.051    TCP    6.6.6.1:993    ->    4.4.4.4:53452        23     4509   196
12:32:51.254   18.040    TCP    6.6.6.1:993    ->    4.4.4.4:53454        23     4509   196
12:33:09.293   11.982    TCP    6.6.6.1:993    ->    4.4.4.4:53456        18     3947   219
```

Listing 6.2: Toky neúspešného útoku - posledný tok je odlišný

Výhodou algoritmu detekcie útokov na SSH je, že dokáže detekovať útoky z viacerých IP adries. Túto vlastnosť som do skriptu neimplementoval z dôvodu časovej náročnosti zmeny a faktu, že všetky detekované útoky v testovacom prostredí prebiehali z jednej IP adresy. V prípade potreby je možné túto vlastnosť do skriptu doplniť.

Skript detekcie útokov na SSH obsahuje dynamické parametre, ktoré sa menia počas behu. Výhodou je, že netreba voliť niektoré parametre pred spustením skriptu, nevýhodou, že skript upravuje hraničné hodnoty podľa každého detekovaného útoku, čo môže ovplyvniť presnosť detekcie. Zároveň môže útočník zneužiť toto správanie a generovaním určitých útokov donútiť algoritmus zmeniť hodnoty podľa potreby.

Detekcia útokov na autentizačné služby (LDAP)

Metrikou na detekciu útoku je veľkosť toku (musí byť menší ako 650 B) alebo dĺžka toku - tok trvá viac ako 20 s. Ak nejaký tok vyhovuje aspoň jednému z týchto pravidiel, je pre zdrojovú IP adresu vypočítaný percentuálny podiel na komunikácii, na základe ktorého je stanica s danou IP adresou označená za útok. Žiadny zo spomenutých parametrov sa v prípade poštových serverov neprejavil ako spoľahlivá charakteristika na detekciu slovníkových útokov.

Detekcia útokov na autentizáciu RDP

Skript detekuje útoky na základe veľkosti a počtu paketov. Ďalšou charakteristikou útokov sú podľa autora TCP príznaky ACK, PUSH, RESET a SYN. Tento predpoklad neplatí pre poštové protokoly. Ako uvádzam v kapitole 4.4, niektoré príznaky (RST) sa v tokoch útokov vyskytujú častejšie, nie je to však pravidlo. Program je vo forme doplnku pre kolektor NfSen - spracováva 5-minútové úseky dát. Na detekciu útoku je potrebné, aby prebehol z danej IP adresy sken siete. Pre slovníkové útoky za poštové servery by bola táto požiadavka

obmedzujúca, pretože IP adresa serveru sa dá získať rôznymi spôsobmi a nie je potrebné skenovať sieť. V reálnom prostredí prebehol sken siete pred samotným slovníkovým útokom asi v 15% prípadov. Následne sú na toky danej IP adresy aplikované filtre s predvolenými veľkosťami paketov a TCP príznakmi, dohľadá sa krajina podľa IP adresy útočníka a zistené informácie sa zapíšu do databázy, aby mohli byť neskôr spracované. Z výsledkov testovania vyplýva, že skript nezachytil všetky útoky na RDP server, pretože nesplňovali charakteristiky útokov, ktoré autor definoval.

Kapitola 7

Zhodnotenie algoritmu a skriptov

Implementované skripty dokazujú, že uvedené spôsoby detekcie a algoritmy je možné použiť na hľadanie slovníkových útokov v laboratórnom aj produkčnom prostredí. Ďalší cieľ bude použiť tieto algoritmy na detekciu útokov v reálnom čase. Táto kapitola popisuje aspekty, ktoré je treba zvážiť pri nasadení uvedených algoritmov na kolektor do reálnej sieťovej infraštruktúry, ako sa dajú použiť výstupy skriptov a aké sú výhody a nevýhody tohto prístupu v porovnaní s existujúcimi riešeniami.

7.1 Detekcia útokov v reálnom čase

Implementované skripty vyhľadávajú útoky vo všetkých tokoch dát za určený časový interval. Ak by sme chceli nepretržite vyhľadávať útoky v aktuálnych tokoch, bolo by časovo náročné a zbytočné opakovane prehľadávať staré toky.

Predpokladajme základné a najbežnejšie nastavenie kolektora nfdump, ktorý vytvára nový záznam o tokoch každých 5 minút. Na detekciu útoku je potrebné zistiť dve hlavné charakteristiky: frekvenciu príchodu tokov a rozptyl určitých parametrov. Prvú charakteristiku je možné detekovať z posledného záznamu. V najhoršom prípade budú toky útoku rozdelené na polovicu medzi dva susedné záznamy. Tento prípad môže robiť problémy pri detekcii krátkych útokov, ale je ho možné vyriešiť znížením minimálnej hranice tokov pre detekciu útoku. Na výpočet rozptylu parametrov potrebuje algoritmus skontrolovať históriu tokov pre danú IP adresu. Skúmanú históriu je možné obmedziť časovo a/alebo počtom tokov. Vďaka nízkej časovej náročnosti algoritmu môžu byť rozptyly parametrov pre danú IP adresu počítané z histórie niekoľkých dní.

7.2 Použitie výstupov, blokovanie útokov

Ak implementovaný skript detekuje útok, zobrazí na štandardný výstup dôležité informácie o danom útoku a útočníkovi. Aké akcie budú na základe daného výstupu vykonané, je na rozhodnutí administrátora. Z výstupov môže byť napríklad vytvorená databáza s informáciami o útokoch. Ďalšou možnosťou je blokovanie útokov. Blokovať sa dá konkrétna IP adresa alebo podsieť, do ktorej IP adresa patrí. Pri detekcii útokov z výstupov kolektora môže robiť problém latencia, s ktorou dostávame nové toky. Toky môžeme spracovať až keď ich sonda odošle na kolektor a v prípade detekcie z nfdump záznamov až po rotácii záznamu. Informáciu o útoku administrátor dostane až po niekoľkých minútach. Pri štandardne nastavenej rotácii nfdump záznamov môžeme predpokladať, že informácia je stará

približne 5 minút. Z grafu 4.1, ktorý zobrazuje dĺžky útokov zachytených v reálnom prostredí je vidieť, že slovníkové útoky trvajú pomerne krátko. 13 útokov z 23 trvalo kratšie ako 5 minút. 8 útokov bolo kratších ako 1 minúta. Z toho vyplýva, že viac ako 50% útokov by nebolo možné zablokovať, pretože by skončili skôr ako by mohli byť detekované. Blokovanie krátkych slovníkových útokov z výstupov nfdump je kvôli veľkému oneskoreniu príchodu tokov v praxi nepoužiteľné.

Aby sme útoky vedeli zablokovať, musíme toky spracovávať skôr. Možným riešením je nečakať na rotáciu záznamov, ale prehľadávať toky prúdovo hneď po ich príchode na kolektor. Týmto prístupom by sa eliminovalo oneskorenie, ktoré spôsobuje rotácia záznamov na kolektore. Aktuálnosť tokov by bola ovplyvnená len časovačmi na sonde a prípadným bufferovaním dát pred odoslaním kolektoru. Keďže útočníci používajú iný port pre jednotlivé pokusy o prihlásenie a po každom pokuse posielajú paket s príznakom FIN 4.4, toky by mali na sonde expirovať okamžite a algoritmus na detekciu útokov by mohol detekovať útok hneď ako sa útočník pokúsi prihlásiť preddefinovaný počet krát.

7.3 Rozšírenie množiny parametrov

Z kapitoly 6.4 vyplýva, že z výstupov kolektora nfdump nie je možné detekovať určité typy slovníkových útokov. Cieľom tejto podkapitoly je určiť, ktoré parametre paketov by pomohli detekcii týchto útokov v prípade, ak by ich sonda exportovala. Parametre paketov som skúmal vo virtuálnom prostredí 4.3.1 s pomocou programov tcpdump[1] a Wireshark[2].

Parametre paketov, ktoré by spresnili alebo umožnili detekciu slovníkových útokov z tokov:

Service Response Time (SRT) - Ak má poštový server implementovanú penalizáciu pri zadaní zlého hesla, útočník musí čakať na odpoveď niekoľko sekúnd. V prípade, že útočník útočí z jedného portu a neukončuje TCP spojenia, vznikajú dlhé toky, z ktorých je obtiažne detekovať útoky. Podľa SRT by sa dalo určiť, či ide o útok. Ak je SRT vyšší ako zvolená hranica a tok obsahuje dostatočný počet paketov, pravdepodobne ide o útok. Vyhovujúcu štatistickú hodnotu pre SRT (priemer/medián...) bude treba určiť testami.

Počet príznakov reset (RST Flag) - Útočník často nastavuje po neúspešnom pokuse o prihlásenie príznak RST. Kolektor nfdump síce zobrazuje nastavené príznaky, ale len spoločne pre všetky pakety v toku. Počet príznakov RST v danom toku by pomohol spresniť detekciu útokov v dlhých tokoch.

Tieto parametre by pomohli detekcii útokov z dlhých tokov, ktoré nebolo možné detekovať z nfdump záznamov. V reálnom prostredí nebol takýto typ útoku zaznamenaný, preto je otázna nutnosť implementácie kvôli dlhým tokom.

Ďalším dôvodom na zavedenie SRT a RST parametrov je **agregácia tokov**. Ak budú toky od útočníka agregované do jedného záznamu, nebude možné vypočítať rozptyly parametrov. Útoky sa budú dať z agregovaných tokov detekovať na základe počtu paketov s príznakom SYN a FIN (útočník často ukončuje a začína nové spojenie). Na spresnenie výsledku sa opäť dajú použiť SRT a RST parametre.

[1] http://www.tcpdump.org/
[2] https://www.wireshark.org/

7.4 Detekcia útokov na iné protokoly

Účel tejto podkapitoly, ktorá mierne vybočuje zo zadania práce je stručne ukázať možnosti navrhnutých algoritmov. Detekčný algoritmus som sa snažil navrhnúť tak, aby nebol viazaný na konkrétny protokol. Jeho testovanie prebiehalo doteraz len na poštových protokoloch, ale mal by dokázať detekovať útoky aj na iné protokoly. Pre overenie tejto hypotézy som otestoval detekčný skript na tokoch zachytených v produkčnom prostredí 4.5. Dataset obsahoval toky zachytené na hraničnom smerovači, predtým, ako bola sonda presunutá na poštový server. Hraničný smerovač neumožňoval detekovať niektoré spätné toky, preto budeme detekovať útoky len z prichádzajúcich tokov, odchádzajúce budú ignorované. Toky boli zbierané po dobu 10 dní.

Útoky na iné protokoly nie je možné klasifikovať pomocou log súborov, pretože by som musel mať prístup ku všetkým serverom v danej sieti. Z útokov na poštové servery v danom dátovom centre vyplýva, že prichádzajú zo zahraničných IP adries. Na klasifikáciu útokov použijem túto znalosť spolu s informáciami z databází Anti-Hacker-Alliance[3], Firyx[4] a Blocklist.de[5].

Detekcia útokov na SSH protokol

Pred detekciou som z tokov zistil, že SSH servery bežia na štandardnom porte 22 a zvolil som tento port v detekčnom skripte. O útokoch na SSH servery nemám žiadne informácie, preto som na skúšku použil hraničné hodnoty nájdené pre protokol POP3 z datasetu č.2. Po spustení našiel skript 159 slovníkových útokov z 93 unikátnych IP adries. Všetky detekované IP adresy sa nachádzali minimálne v jednej z troch databází nebezpečných IP adries. Skript nezaznamenal žiadnu falošnú pozitívnu detekciu. Pri ďalšom pokuse som hranice zdvihol 10-násobne. Skript detekoval 190 útokov zo 110 unikátnych adries. Po porovnaní s databázami nebola zaznamenaná žiadna falošná detekcia. Výsledky všetkých testov pre zvolené hraničné hodnoty zobrazuje tabuľka 7.1.

Číslo testu	**Maximálny rozptyl**		**Unikátne IP adresy útočníkov**	**Detekované útoky**	**Falošné detekcie**
	bajty/paket	**počet paketov**			
1	12	80	93	159	0
2	120	800	110	190	0
3	1200	8000	251	389	0
4	12000	80000	276	420	1

Tabuľka 7.1: Detekované útoky na SSH servery

Z výsledkov je vidieť, že na detekciu SSH útokov sme museli nastaviť veľký rozptyl paketov. Útoky na SSH servery majú vyššiu frekvenciu príchodu paketov ako útoky na poštové servery, preto sa dajú ľahšie rozlíšiť od legitímnej prevádzky. Pri poslednom teste pripadla na 276 útokov z unikátnych IP adries len 1 falošná detekcia.

Detekcia útokov na LDAP a RDP protokol

Okrem SSH som otestoval skripty na LDAP a RDP protokoloch. Pre LDAP som detekoval útoky na portoch 389 a 636. Skript našiel 5 útokov z dvoch IP adries. Vzhľadom na nízky

[3] http://anti-hacker-alliance.com/
[4] https://www.firyx.com/
[5] http://www.blocklist.de/

počet tokov (286 v jednom smere) som mohol overiť, že sa okrem týchto dvoch detekovaných útokov nenachádza medzi danými tokmi žiadna ďalšia podozrivá IP adresa. Obe detekované IP adresy patrili do rovnakej podsiete, jednotlivé útoky trvali približne 30 sekúnd a obsahovali 89 tokov. Rozptyl paketov bol 0.045 a bajtov na paket 0.0 v oboch prípadoch.

Tokov RDP bolo narozdiel od LDAP veľa, preto výsledky testovania zhrniem do tabuľky 7.2.

Číslo testu	**Maximálny rozptyl**		**Unik. IP adresy útočníkov**	**Detekované útoky**	**Falošné detekcie**
	bajty/paket	**počet paketov**			
1	6	40	49	149	0
2	12	80	53	165	0
3	120	800	53	165	0
4	120000	800000	53	165	0

Tabuľka 7.2: Detekované útoky na RDP servery

Z tabuľky je vidieť, že zvyšovanie hraníc nad určitú hodnotu nemalo vplyv na množstvo detekovaných útokov, z čoho vyplýva, že útoky na RDP by sa v tomto prípade dali detekovať len na základe frekvencie prichádzajúcich tokov. Štyri detekované IP adresy som nenašiel v databázach útočníkov, ale keďže pochádzali zo zahraničia, je pravdepodobnosť, že išlo o útok veľmi vysoká.

Detekcia útokov na ostatné protokoly

Služieb na ktorých by sme mohli testovať algoritmus je veľké množstvo. Nie je časovo možné otestovať všetky, preto som ďalší test spravil bez nastavenia konkrétnych portov. Minimálny počet tokov je 20, frekvencia 5 tokov za sekundu, rozptyl počtu paketov 30 a rozptyl bajtov na paket 100. IP adresy útočníkov boli overované v databázach rovnako ako v predchádzajúcich testoch. Skript detekoval útoky na nasledujúce služby:

- SIP server (port 5060)
- HTTP a HTTPS formuláre (port 80 a 8080)
- MySQL a MSSQL server (port 1433, 3306)
- Microsoft DS, Active Directory (port 445)
- NetBios (port 139)
- Telnet (port 23)
- a ďalšie.

Okrem slovníkových útokov je možné detekovať aj ďalšie typy útokov, ktoré generujú veľmi podobné toky. Takýmto útokom je napríklad **skenovanie siete**. Detekčný skript zachytil vertikálne aj horizontálne skenovania siete. Pri horizontálnom skenovaní siete útočník posiela ICMP pakety na rôzne IP adresy z danej podsiete, čím sa snaží nájsť aktívne zariadenia. Po nájdení aktívnych zariadení väčšinou nasleduje vertikálne skenovanie, ktorým útočník zisťuje otvorené porty a služby na danom zariadení. Pre obidva typy skenovania platí, že toky od útočníka majú vysokú frekvenciu príchodov, sú veľmi krátke a obsahujú

rovnaký počet paketov a bajtov. Tieto parametre sú podobné charakteristikám slovníkových útokov v tokoch, preto je ich možné jednoducho detekovať vytvoreným skriptom. Horizontálne skenovania môže skript detekovať vďaka tomu, že mu nezáleží na cieľovej IP adrese obete.

Z výsledkov testov pre SSH, LDAP, RDP a iné protokoly môžeme vyvodiť záver, že implementovaný skript spoľahlivo detekuje slovníkové útoky na rôzne typy serverov. Na upresnenie výsledkov by som potreboval prístup k log súborom daných serverov, aby bolo možné zistiť presný počet skutočných útokov, ale takáto podrobná analýza nie je predmetom tejto práce. Okrem slovníkových útokov dokáže skript detekovať napríklad skenovanie siete a iné útoky založané na opakovanom vykonávaní určitej akcie.

Kapitola 8

Záver

Z predchádzajúcich výskumov, ktoré som v práci popísal vyplýva, že je možné detekovať slovníkové útoky z tokov dát. Dôkazom je napríklad implementácia detekcie týchto útokov na protokol SSH. Aby som overil, či platia deklarované prístupy aj pre poštové protokoly, bolo vytvorené testovacie prostredie s poštovým serverom, sondou a kolektorom. Útoky na toto prostredie boli generované nástrojmi Hydra a Medusa s rôznymi parametrami (počet súbežných spojení, počet hesiel v slovníku, typ zabezpečenia, intervaly medzi jednotlivými pokusmi...). Zo získaných výsledkov som vyvodil charakteristiky tokov pre útoky na poštové protokoly:

- veľký počet tokov a paketov za určitý časový interval
- malý rozptyl počtu paketov v tokoch patriacich konkrétnemu útoku
- malý rozptyl priemerného počtu bajtov na paket
- prvý a posledný tok sa možu líšiť od ostatných tokov
- dĺžka tokov je rôzna a závisí od penalizácie a počtu pokusov o prihlásenie v jednom toku

Namiesto ostrých hraníc pre počet paketov v toku a dĺžku tokov som sa rozhodol použiť pre detekciu útokov rozptyl parametrov - konkrétne počet bajtov na paket a počet paketov v toku. Táto metrika je menej závislá na konkrétnej implementácii poštového serveru a útočníkovej aplikácii. Aj keď sú charakteristiky tokov pre jednotlivé útoky odlišné, rozptyl parametrov jednotlivých tokov zostáva veľmi malý, pretože útoky sú generované programami, ktoré vykonávajú tú istú operáciu (pokus o prihlásenie) pre každé heslo zo slovníka. Rozptyl parametrov tokov, ktoré vznikli pri akciách legitímneho používateľa, je naopak veľký, pretože používatelia prevádzajú rozličné operácie, ktoré generujú heterogénne charakteristiky tokov. Aby bolo možné zistené charakteristiky overiť v reálnom prostredí, sonda a nfdump kolektor boli nasadené do serverovne menšieho poskytovateľa internetu a hostingových služieb. Toky som klasifikoval na základe log súborov z poštového serveru. Charakteristiky zachytených dát som porovnal s charakteristikami vytvorenými vo virtuálnom prostredí a na ich základe som implementoval skript v jazyku Python, ktorý detekuje slovníkové útoky z výstupov nfdump kolektora.

Citlivosť detekcie ovplyvňujú hraničné hodnoty parametrov, ktoré musia byť zvolené pred spustením detekčného skriptu. Aby nemusel administrátor tieto hranice zisťovať ručne, bol vytvorený skript, ktorý ich vypočíta zo známych IP adries útokov. Vypočítané hranice

sú maximálne hodnoty rozptylov, ktoré by mal administrátor následne upraviť, aby sa dosiahla optimálna miera detekcie.

Skript som otestoval vo virtuálnom aj reálnom prostredí. V reálnom prostredí dosiahol úspešnosť detekcie 100% a nespôsobil žiadnu falošnú detekciu. Vo virtuálnom prostredí sa mi podarilo vygenerovať útoky, ktoré nie je možné detekovať z výstupov nfdump. Ak útočník nemení po každom pokuse o prihlásenie port, viacero pokusov vygeneruje jediný tok. Ak je útok veľmi krátky, vygenerovaných tokov je príliš málo na to, aby sa dalo spoľahlivo určiť, či sa jedná o útok. V niekoľkých prípadoch sa celý útok 'zmestil' do jedného toku. Na detekciu tohto typu útokov by museli byť do sondy pridané IE SRT a počet SYN príp. RST príznakov v tokoch.

Okrem detekcie útokov bude často vyžadované ich zablokovanie. Z charakteristických znakov správania útočníkov, ktoré som zistil z dát zachytených v reálnom prostredí vyplýva napríklad, že slovníkové útoky sú krátke: 50% zachytených útokov trvalo kratšie ako 5 minút, čo obmedzuje možnosti blokovania útokov z výstupov nfdump kolektora. Krátke útoky by nebolo možné zablokovať, pretože by skončili skôr, ako by mohli byť detekované. Z tohto dôvodu je potrebné implementovať blokovanie slovníkových útokov skôr ako sú spracované kolektorom nfdump. Pri prúdovom spracovaní tokov by mohli byť útoky spoľahlivo zablokované vďaka faktu, že útočník mení port po každom pokuse o prihlásenie a sonda vytvorí pre každý pokus nový tok.

Existujúce práce, ktoré sa zaoberajú detekciou slovníkových útokov používajú na odlíšenie útokov ostré hranice veľkosti a počtu paketov. Takýto prístup môže fungovať pri použití konkrétnych aplikácií v konkrétnom prostredí. Poštových protokolov a serverov je ale veľké množstvo a charakteristiky útokov sa menia v závislosti na použitom softvéri obete aj útočníka, preto som sa snažil, aby bola navrhnutá metóda čo najmenej závislá na prostredí, v ktorom bude použitá. Výsledkom je algoritmus, ktorého použitie nie je závislé len na poštových serveroch. Uskutočnené testy potvrdzujú, že pomocou algoritmu môžeme detekovať útoky na rôzne služby. Ak by sme nekontrolovali toky len pre konkrétny port, je možné zachytiť horizontálne a vertikálne skenovanie siete. Pri reálnom nasadení by nebolo nutné používať rozdielne algoritmy pre SSH, RDP alebo LDAP, ako je to v prípade predchádzajúcich výskumov, detekcia by sa líšila len zvolenými hraničnými hodnotami pre daný protokol.

Literatúra

[1] BROWNLEE, N.: Flow-Based Measurement: IPFIX Development and Deployment. In Proceedings of IEICE Transactions, 2011, pp.2190-2198.

[2] CLAISE, B.: Cisco Systems NetFlow Services Export Version 9. Technická zpráva, RFC 3954, IETF, 2004.

[3] CLAISE, B.: Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of IP Traffic Flow Information. Technická zpráva, RFC 5101, IETF, 2008.

[4] CRISPIN, M.: INTERNET MESSAGE ACCESS PROTOCOL - VERSION 4rev1. Technická zpráva, RFC 3501, IETF, 2003.

[5] FastMail: SSL vs TLS vs STARTTLS.
URL https://www.fastmail.com/help/technical/ssltlsstarttls.html

[6] GELLENS, R.; KLENSIN, J.: Message Submission. Technická zpráva, RFC 2476, IETF, 1998.

[7] HOFSTEDE, RICK and ČELEDA, PAVEL and TRAMMELL, BRIAN and DRAGO, IDILIE and SADRE, RAMIN and SPEROTTO, ANNA and PRAS, AIKO: Flow Monitoring Explained: From Packet Capture to Data Analysis with NetFlow and IPFIX. *IEEE Communications Surveys & Tutorials*, 2014, ISSN 1553-877X.

[8] INVEA-TECH: FlowMon Collector Models List. Technická zpráva, INVEA-TECH, 2015 [cit. 2015-01-03].
URL https://www.invea.com/data/flowmon/flowmon_collector_specification_en.pdf

[9] J., V.; PLESNIK, T.; MINARIK, P.: Network-Based Dictionary Attack Detection. In *Future Networks, 2009 International Conference on*, March 2009, s. 23–27.

[10] J. QUITTEK, T. ZSEBY, B. CLAISE: Requirements for IP Flow Information Export (IPFIX). Technická zpráva, RFC 3917, IETF, 2004.

[11] JOSEFFSON, S.: The Base16, Base32, and Base64 Data Encodings. Technická zpráva, RFC 4648, IETF, 2006.

[12] KLENSIN, J.: Simple Mail Transfer Protocol. Technická zpráva, RFC 5321, IETF, 2008.

[13] LEINEN, S.: Evaluation of Candidate Protocols for IP Flow Information Export (IPFIX). Technická zpráva, RFC 3955, IETF, 2004.

[14] MYERS, J.: Simple Authentication and Security Layer (SASL). Technická zpráva, RFC 2222, IETF, 1997.

[15] MYERS, J.; ROSE, M.: Post Office Protocol - Version 3. Technická zpráva, RFC 1939, IETF, 1996.

[16] PINKAS, B.; SANDER, T.: Securing Passwords Against Dictionary Attacks. In *Proceedings of the 9th ACM Conference on Computer and Communications Security*, CCS '02, New York, NY, USA: ACM, 2002, ISBN 1-58113-612-9, s. 161–170. URL http://doi.acm.org/10.1145/586110.586133

[17] SATOH, A.; NAKAMURA, Y.; IKENAGA, T.: SSH Dictionary Attack Detection Based on Flow Analysis. In *Applications and the Internet (SAINT), 2012 IEEE/IPSJ 12th International Symposium on*, July 2012, s. 51–59.

[18] SIEMBORSKI, R.; MENON-SEN, A.: The Post Office Protocol (POP3) Simple Authentication and Security Layer (SASL) Authentication Mechanism. Technická zpráva, RFC 5034, IETF, 2007.

[19] SPEROTTO, A.; SCHAFFRATH, G.; SADRE, R.; aj.: An Overview of IP Flow-Based Intrusion Detection. *Communications Surveys Tutorials, IEEE*, ročník 12, č. 3, Third 2010: s. 343–356, ISSN 1553-877X, doi:10.1109/SURV.2010.032210.00054.

[20] VIZVÁRY, M.: *Detekce útoků na autentizaci RDP*. Diplomová práce, Masarykova univerzita, Fakulta informatiky, 2013 [cit. 2015-01-30].

[21] VYKOPAL, J.; PLESNÍK, T.; MINAŘÍK, P.: Validation of the Network-based Dictionary Attack Detection. In *Security and Protection of Information 2009, Proceeding of the Conference*, University of Defence, 2009, ISBN 978-80-7231-641-0, s. 128–136.

[22] ŠEMBERA, R.: *Detekce slovníkových útoků na autentizační služby*. Diplomová práce, Masarykova univerzita, Fakulta informatiky, 2010 [cit. 2015-02-03].

Zoznam skratiek

IDS Intrusion Detection System

IE Informačný Element

IETF Internet Engineering Task Force

IMAP Internet Message Access Protocol

IPFIX Internet Protocol Flow Information Export

IPS Intrusion Prevention System

LDAP Lightweight Directory Access Protocol

MSA Mail Submission Agent

MTA Mail Transfer Agent

MUA Mail User Agent

POP3 Post Office Protocol

SASL Simple Authentication and Security Layer

SMTP Simple Mail Transfer Protocol

SRT Service Response Time

Dodatok A

Obsah CD

A.1 Skript dictatt

`[Jednotka CD-ROM]/dictAtt`

A.2 Skript learnchar

`[Jednotka CD-ROM]/learnChar`

A.3 Technická správa

`[Jednotka CD-ROM]/xcinca00.pdf`

A.4 Technická správa - zdrojové texty

`[Jednotka CD-ROM]/ThesisSrc`

Dodatok B

Útok na Microsoft Exchange Server

Listing B.1: Útok na Microsoft Exchange Server (prichádzajúce toky)

```
0  First seen    Duration Src Addr:Port          Dst IP:Port          Flags Pac  Bytes  pps   bps  Bpp
1  16:18:39.851  0.149    6.6.6.1:45164 ->      4.4.4.4:993          .AP.SF 13   1702    87   91382 130
2  16:18:39.872  0.124    6.6.6.1:45166 ->      4.4.4.4:993          .AP.SF 13   1702   104  109806 130
3  16:18:39.827  0.170    6.6.6.1:45162 ->      4.4.4.4:993          .AP.SF 13   1702    76   80094 130
4  16:18:39.804  0.246    6.6.6.1:45160 ->      4.4.4.4:993          .AP.SF 18   2360    73   76747 131
5  16:18:39.688  0.183    6.6.6.1:45150 ->      4.4.4.4:993          .AP.SF 18   2344    98  102469 130
6  16:18:39.719  0.234    6.6.6.1:45153 ->      4.4.4.4:993          .AP.SF 18   2328    76   79589 129
7  16:18:39.993  0.065    6.6.6.1:45175 ->      4.4.4.4:993          .AP.SF 12   1666   184  205046 138
8  16:18:39.965  0.058    6.6.6.1:45173 ->      4.4.4.4:993          .AP.SF 12   1666   206  229793 138
9  16:18:39.740  0.225    6.6.6.1:45155 ->      4.4.4.4:993          .AP.SF 18   2344    80   83342 130
10 16:18:39.912  0.087    6.6.6.1:45169 ->      4.4.4.4:993          .AP.SF 12   1650   137  151724 137
11 16:18:39.791  0.072    6.6.6.1:45159 ->      4.4.4.4:993          .AP.SF 17   2292   236  254666 134
12 16:18:39.766  0.211    6.6.6.1:45157 ->      4.4.4.4:993          .AP.SF 18   2344    85   88872 130
13 16:18:39.936  0.077    6.6.6.1:45171 ->      4.4.4.4:993          .AP.SF 12   1666   155  173090 138
14 16:18:39.728  0.171    6.6.6.1:45154 ->      4.4.4.4:993          .AP.SF 18   2344   105  109660 130
15 16:18:39.953  0.064    6.6.6.1:45172 ->      4.4.4.4:993          .AP.SF 12   1666   187  208250 138
```

Listing B.2: Útok na Microsoft Exchange Server (odchádzajúce toky)

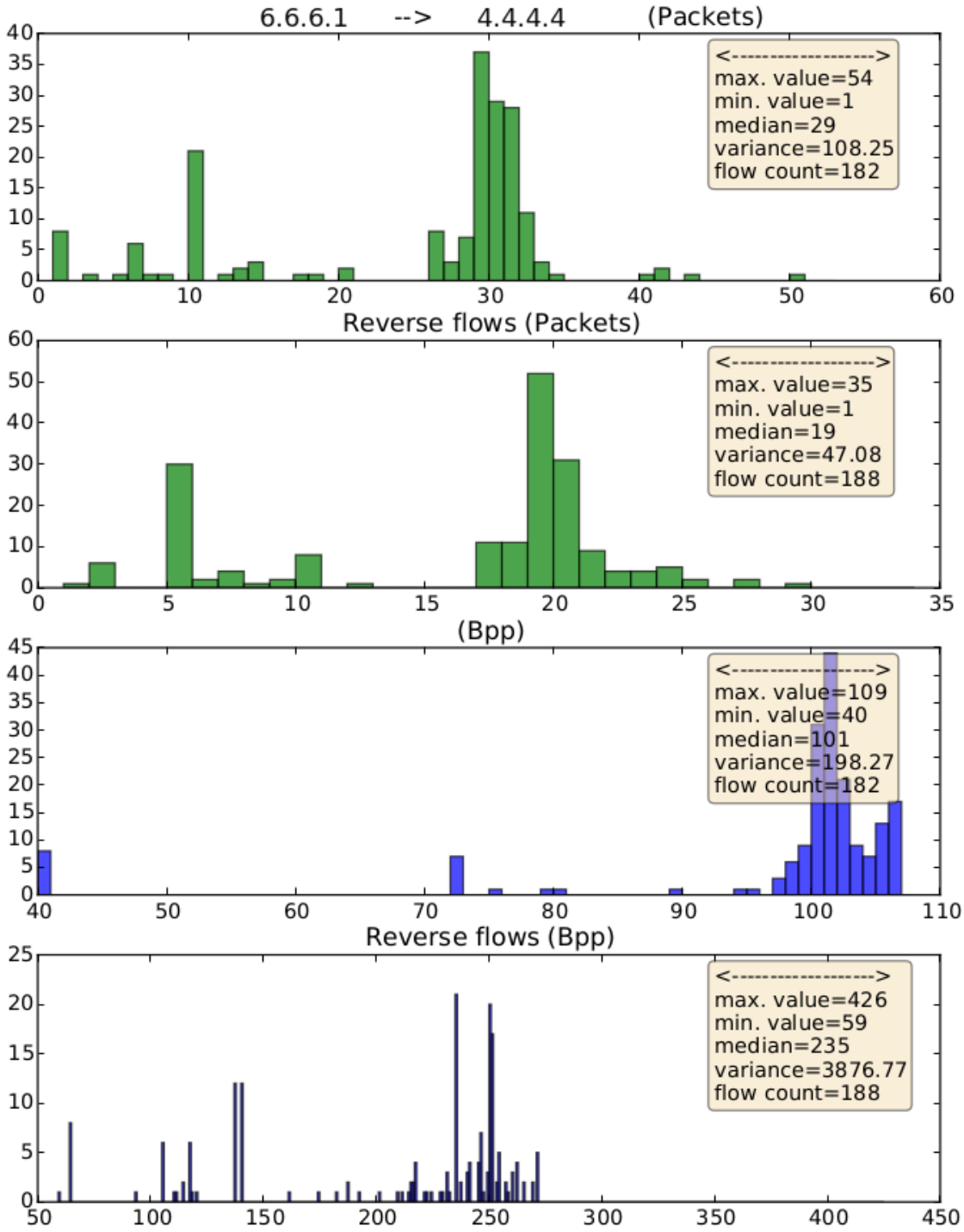
```
0  First seen    Duration Src Addr:Port          Dst IP:Port          Flags Pac  Bytes  pps   bps  Bpp
1  16:18:39.901  0.097    4.4.4.4:993 ->        6.6.6.1:45168        .AP.SF  9   1768    92  145814 196
2  16:18:39.754  0.160    4.4.4.4:993 ->        6.6.6.1:45156        .AP.SF 15   2430    93  121500 162
3  16:18:39.925  0.075    4.4.4.4:993 ->        6.6.6.1:45170        .AP.SF  9   1768   120  188586 196
4  16:18:39.740  0.226    4.4.4.4:993 ->        6.6.6.1:45155        .AP.SF 15   2430    66   86017 162
5  16:18:39.966  0.057    4.4.4.4:993 ->        6.6.6.1:45173        .AP.SF  9   1768   157  248140 196
6  16:18:39.805  0.245    4.4.4.4:993 ->        6.6.6.1:45160        .AP.SF 16   2450    65   80000 153
7  16:18:39.886  0.111    4.4.4.4:993 ->        6.6.6.1:45167        .AP.SF  9   1768    81  127423 196
8  16:18:39.689  0.183    4.4.4.4:993 ->        6.6.6.1:45150        .AP.SF 15   2430    81  106229 162
9  16:18:39.863  0.131    4.4.4.4:993 ->        6.6.6.1:45165        .AP.SF  9   1768    68  107969 196
10 16:18:39.828  0.169    4.4.4.4:993 ->        6.6.6.1:45162        .AP.SF  9   1768    53   83692 196
11 16:18:39.842  0.157    4.4.4.4:993 ->        6.6.6.1:45163        .AP.SF  9   1768    57   90089 196
12 16:18:39.852  0.148    4.4.4.4:993 ->        6.6.6.1:45164        .AP.SF  9   1768    60   95567 196
13 16:18:39.700  0.237    4.4.4.4:993 ->        6.6.6.1:45151        .AP.SF 15   2430    63   82025 162
14 16:18:39.873  0.123    4.4.4.4:993 ->        6.6.6.1:45166        .AP.SF  9   1768    73  114991 196
15 16:18:39.817  0.179    4.4.4.4:993 ->        6.6.6.1:45161        .AP.SF 11   1978    61   88402 179
16 16:18:39.667  0.326    4.4.4.4:993 ->        6.6.6.1:45149        .AP.SF 15   2430    46   59631 162
```

Dodatok C

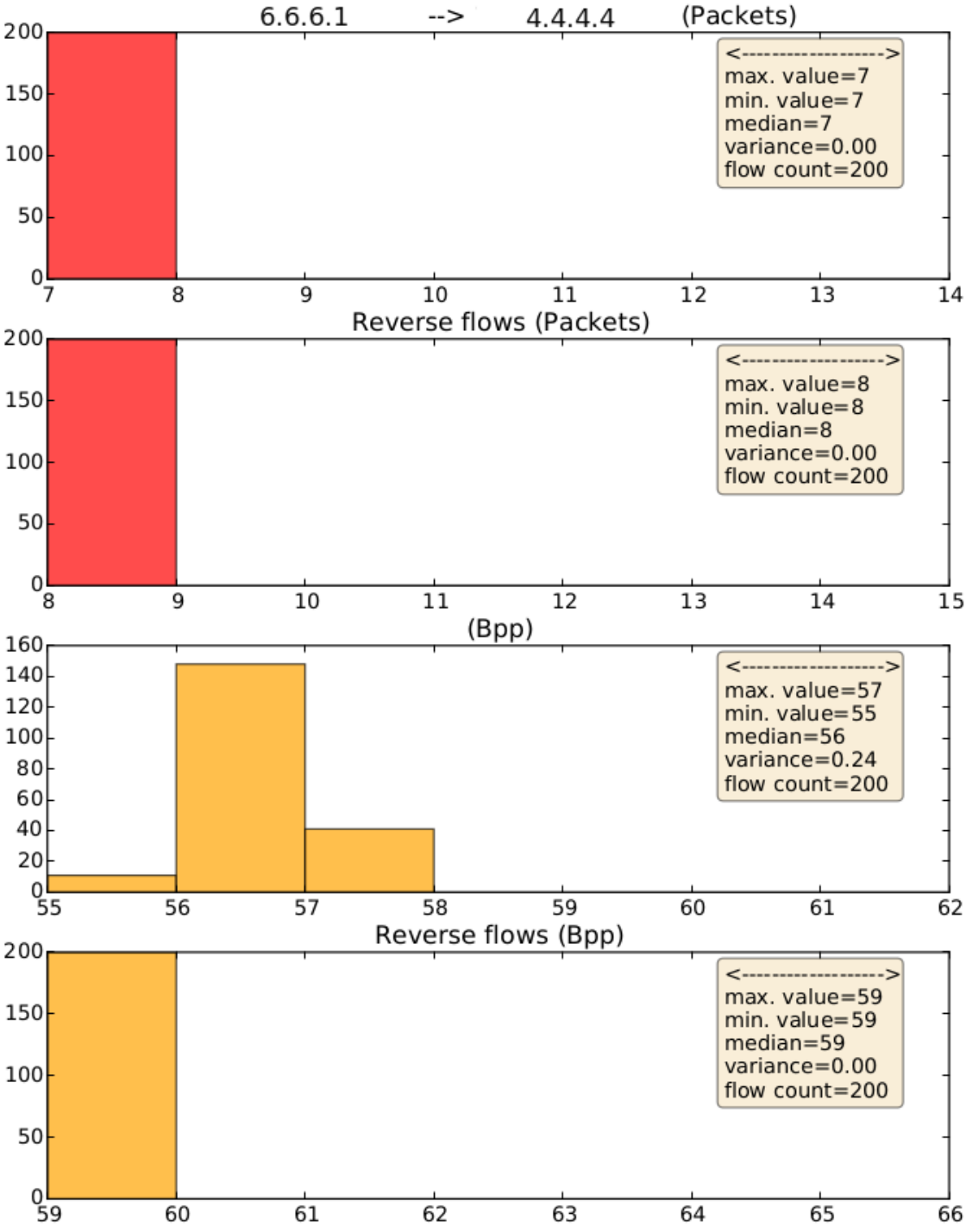
Histogramy vygenerované skriptom

Skript na detekciu útokov po spustení s parametrom –graphs vygeneruje histogramy pre jednotlivé IP adresy v datasete. Počet IP adries, pre ktoré sú histogramy generované je obmedzený kvôli časovej náročnosti generovania histogramov na 10 legitímnych IP adries a všetky IP adresy útočníkov. Pre každú zvolenú IP adresu sa vygenerujú 4 histogramy: 2 histogramy ktoré zobrazujú počet paketov v tokoch (1. pre prichádzajúce a 2. pre odchádzajúce toky) a 2 histogramy zobrazujúce priemerný počet bajtov na paket.

Histogramy, ktoré zobrazujú toky útočníkov sú odlíšené oranžovou a červenou farbou.



Obr. C.1: Histogram pre IP adresu - nebol detekovaný útok



Obr. C.2: Histogram pre IP adresu - útok